AD___________________

Award Number: DAMD17–03–2-0001

TITLE: Advanced Technologies in Safe and Efficient Operating Rooms

PRINCIPAL INVESTIGATOR: Adrian E. Park, M.D.

CONTRACTING ORGANIZATION: University of Maryland Medical Center
Baltimore, Maryland

REPORT DATE: February 2008

TYPE OF REPORT: Annual

PREPARED FOR: U.S. Army Medical Research and Materiel Command
Fort Detrick, Maryland 21702-5012

DISTRIBUTION STATEMENT: Approved for Public Release;
Distribution Unlimited

The views, opinions and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy or decision unless so designated by other documentation.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)* 01-02-2008	**2. REPORT TYPE** Annual	**3. DATES COVERED** *(From - To)* 1 Feb 2007 - 31 Jan 2008

4. TITLE AND SUBTITLE

Advanced Technologies in Safe and Efficient Operating Rooms

5a. CONTRACT NUMBER

5b. GRANT NUMBER
DAMD17–03–2-0001

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Adrian E. Park, M.D.

E-Mail: gmoses@smail.umaryland.edu

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

University of Maryland Medical Center
Baltimore, Maryland

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
U.S. Army Medical Research and Materiel Command
Fort Detrick, Maryland 21702-5012

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for Public Release; Distribution Unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

The three major research targets of this study are (a) OR informatics, (b) simulation research, and (c) smart image. The purpose of the OR informatics program is to develop, test, and deploy technologies to collect real-time data about key tasks and process elements in clinical operating rooms. The objective of Simulation research is to create a system where a user can interact with a virtual human model in cognitive simulation and have the virtual human respond appropriately to user queries and interventions in clinical situations, with a focus on cognitive decision making and judgment. The objective of smart image is to use real-time 3D ultrasonography and 40-slice high-framerate computed tomography (CT) for intraoperative imaging to volume rendered anatomy from the perspective of the endoscope. The overall project reported here has proceeded under the mantle of "Operating Room of the Future" research. We are replacing that theme with the more appropriate "Innovations in the Surgical Environment". The period of performance of this contract was extended to February 28, 2009. Based on the extended period of performance, this project is on time, on schedule, and within performance parameters.

15. SUBJECT TERMS
No subject terms were provided.

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON** USAMRMC
a. REPORT U	**b. ABSTRACT** U	**c. THIS PAGE** U	UU	179	**19b. TELEPHONE NUMBER** *(include area code)*

Table of Contents

1. Current staff, role and percentage effort on each project.

Staff Member	Role
A. Park, MD	PI
G. Moses, PhD.	Director
C. MacKenzie, MD	IT-Coord
B. Jarrell, MD	Co-PI
D. Mallott, MD	Researcher
G. Fantry, MD	Clinical Support
R. Shekhar, PhD	Smart Image
Post Doc	Smart Image
W. Berntstein, MD	Smart Image

Introduction

During the past decade, we witnessed an extraordinary evolution in surgical care based upon rapid advances in technology and creative approaches to medicine. The increased speed and power of computer applications, the rise of visualization technologies related to imaging and image guidance, improvement in simulation-based technologies (tissue properties, tool-tissue interaction, graphics, haptics, etc) has caused an explosion in surgical advances. That said, we remain far behind scientists in applying information systems to patient care. This research effort has proceeded under the mantle of "Operating Room of the Future" research. We are replacing that theme with the more appropriate "Innovations in the Surgical Environment."

The content of this annual report contains information pertinent to continued activities in relation to the DAMD-17-03-2-0001 (Modified) "Advanced video technology for safe and efficient surgical operating rooms" project ("ORF-Y3"). This contract consists of a scope of work that fits seamlessly onto the current continuing research and activity in the contract W81XWH-06-2 "Advanced technologies in safe and efficient operating rooms" work ("ORF-Y4"). The ORF-Y3 activities are based upon three pillars of research, OR Informatics, Simulation for Training and Smart Image. A fourth pillar will be added to future research endeavors, cognitive ergonomics/human factors. Objectives listed in this annual report are aligned with current (Y4) objectives.

There are five projects that comprise the Informatics pillar and two for Smart Image. The Simulation pillar has been comprised of a single project, The Maryland Virtual Patient. This pillar will be expanded to include research conducted in and for the larger Simulation Training Program in the MASTRI Center.

Body

A. OR Informatics

Informatics subgroup 1. The Perioperative Scheduling Study (WORQ)

The Perioperative Scheduling Study is looking at how using post-operative destination information during the process of surgery scheduling can influence congestion in post-operative units such as ICUs and IMCs, which lead to overnight boarders in the PACU. The research team is Jeffrey W. Herrmann, Ph.D., and Greg Brown, a graduate student, both with the University of Maryland, College Park. The team is working closely with Michael Harrington, Ramon Konewko, R.N., and Paul Nagy, Ph.D., for guidance and assistance.

The surgery scheduling process has been carefully studied to understand the different organizations and persons who participate in the process, include the schedulers in the surgical services, the perioperative services office, the PACU manager, and the OR charge nurse. Interviews with many of these groups and observations of their scheduling process were conducted on January 17, 2008. These groups also provided copies of their scheduling policies and typical schedules.

We are currently developing a mathematical model for evaluating congestion in post-operative units, including ICUs, IMCs, and floor units. This model will require data about post-operative destinations and length-of-stay distributions for different types of surgeries. Currently, data about cardiac surgeries from two years is being analyzed to develop a methodology for computing the needed information. This methodology will be applied to a larger set of historical data to generate a complete set of information to execute the congestion evaluation model.

Informatics subgroup 2. Operating Room Glitch Analysis (OGA)

The OGA project, focusing on institutional learning, is looking at the workflow around performance indicators in the perioperative environment and building a graphical dashboard to allow data mining and trend analysis of operating indicators.

The dashboard was constructed using Ruby on Rails web development platform with a MySQL database dynamically driving the queries. An interactive graphical dashboard provided synthesis around delays in operations with multiple information visualization techniques.

The web site was demonstrated at the annual operating room of the future conference in July of 2007 in Columbia Maryland. A manuscript was submitted and accepted into the Journal of Surgical Innovation with a March publication date of 2008. Attached to this annual report is the final accepted manuscript which provides an in depth description of the project.

The web site project was a combination of operations research metrics along with information visualization and business analytics. Clinical validation was done with Surgery, Anesthesia as well as the Chief Medical Officer, Chief Nursing Officer, and Chief Operating Officer at the University of Maryland Medical Center.

The clinical surgical informatics team led by Ramon Konewko RN has taken over support and continued development of the analytics tool developed as part of this project.
.

Informatics subgroup 3. Context Aware Surgical Training (CAST)

We proposed to design and implement a prototype context aware surgical training environment (CAST) as part of the University of Maryland Medical System’s SimCenter. This system will be used to explore the role that an intelligent pervasive computing environment can play to enhance the training of surgical students, residents and specialists. The research will build on prior work on context aware “smart spaces” done at UMBC; leverage our experience in working with RFID in the DARPA Trauma Pod program as well as in incorporating Web-based infrastructure and software applications in academic and professional development programs. The project will result in pilot system integrating one or two training resources available in the SimCenter into a context aware training environment that can recognize the presence of a trainee and or mentor and take appropriate action based on known training goals and parameters. The project will advance the knowledge of context aware training environments in a highly technical medical field and provide a basis for incorporating more advanced technology assisted learning experiences in medicine. This “smart environment” may then, if successful, be scaled to meet the needs of an operative environment where the technological demands may be the similar or analogous to those seen in the training environment. A goal of this project is to advance interactive information, resource, and content management via a seamless process. Ultimately, the advanced training and potential for use in perioperative environments have a long-term end goal of improving patient safety and adding to the body of knowledge in surgical training. Initially, we see a situation were clinicians in training can receive a tailored curriculum. Additionally, we envision a system that offers real-time feedback and decision support and education metrics to faculty.

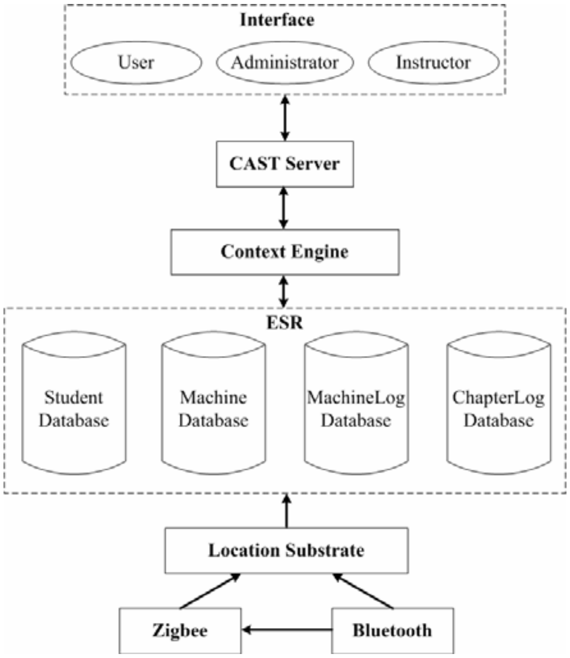
A key goal this year was to prototype the CAST system. To start off, we had meetings with the MASTRI team responsible for the training efforts led by Dr Turner to iron out the requirements for the system and came up with the following (initial) set of tasks to be accomplished

- Student Tracking
- Enforcing Prerequisites
- Video Capture
- Instructor Feedback

Also, we defined a typical use case for our system.
A Student enters the simulation center. The system identifies the student (for instance, using their Bluetooth phone or their badge), and does a prerequisite check based on the simulator the student wants to perform the procedure. Only if the student is done with the prerequisites, is he/she allowed to proceed. When the student indicates that they are ready to begin, the system starts capturing the external and internal view until the student indicates that they have completed the task. The captured video is then transferred to the video server for review by the instructor. The instructor interface allows the instructor to see the entry logs of students in terms of when they entered and exited the centre along with the corresponding external view.

We employed the spiral prototyping approach as an experimental test bed; we designed and implemented an initial system prototype that would meet the above functional requirements. The prototype integrates two machines with each simulator -- a small Nokia 800 device for resident interaction, and a larger PC for video capture. Note that this is for the proof of concept. A single small form factor but computationally powerful machine could be used instead. In fact, for simulators such as the VR, we expect that eventually manufacturers could integrate our system directly into the computer that drives the simulation.

Our prototype used Bluetooth for localization of residents in the simulation centre. It was designed to be modular, so that any other technology (such as resident ID cards) could be integrated easily. We also hosted training materials including videos for FLS, Kentucky and Rosser tasks in our system, and tracked student progress through the chapters checked out. This was used for enforcing prerequisites when students entered the simulation centre to perform procedures. In addition to enforcing prerequisites, there was a need for the instructors to visually see what the residents were doing during their simulation procedures. We use N800's built in camera to capture the residents' external views. These video feeds are then fed into a central server for review by the instructor.



For location detection, we also experimented with using the Awarepoint tags. Awarepoint uses a zigbee based mesh network for localization and exposes the location information through a web service. Our experiments indicated that Awarepoint could provide us room level information, but not anything finer. While this would help identify if the residents were in the simulation center, it would not help determine which machine they were using, which was needed for CAST. We demonstrated our first system prototype at the ORF workshop by going through a typical student workflow.

Based on feedback on individual components of the first prototype, we started the second version of the prototype to be deployed at the MASTRI center. The key changes in the second prototype from the first one are described below.

- We no longer use awarepoint for locationing since it could provide us only room level accuracy.
- On the student identification front, we are using a standard username and password method for now. Also, since we have external camera views from the

N800, students identified can be verified during the review process by the instructor. Bluetooth based identification exists, but is not used since we were told that most residents may not have phones with Bluetooth. Multiple candidate technologies for identification such as Bluetooth, RFID, nearfield RF badges etc. have emerged and been discussed with MASTRI staff. No single choice has been made yet – the idea is to first make the system robust from a use perspective, and then integrate identification technologies based on further discussion with MASTRI staff. Our system is capable and flexible enough to handle a variety of lower level locationing technologies and therefore we would choose the one that is most practical in MASTRI scenario.

- Due to hospital network firewall policies, we had to move away from using a wireless network for transferring videos from the N800 to the MASTRI video server. We currently achieve this by tunneling through the internal view capturing machine which is hooked to the N800 by a USB cable.
- Prerequisite checks are temporarily suspended since the initial classes being taught in MASTRI are not following FLS.

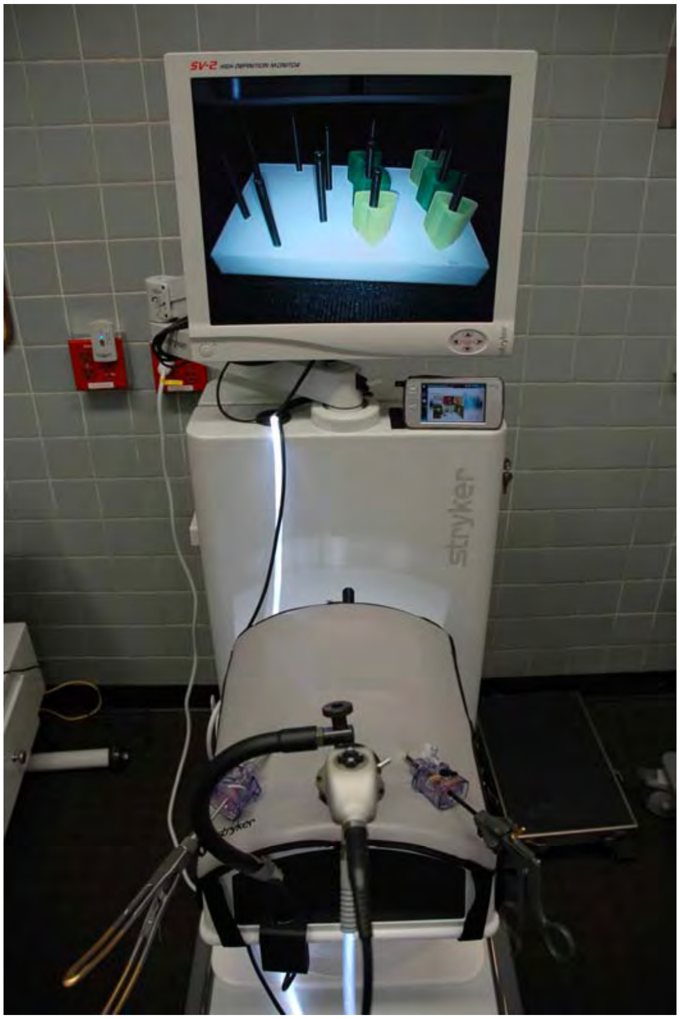
We also focused on moving the system from UMBC machines to the MASTRI infrastructure where they will be housed. We purchased a small factor Dell machine to be used for capturing internal views from simulators. Storage was purchased and added to the mastri-internal server for archiving both internal and external video feeds with help from Jesus. Also, we have

- Integrated the student database from the hospital
- Hosted FLS and other training videos on the hospital infrastructure
- Hacked internal views of the simulators

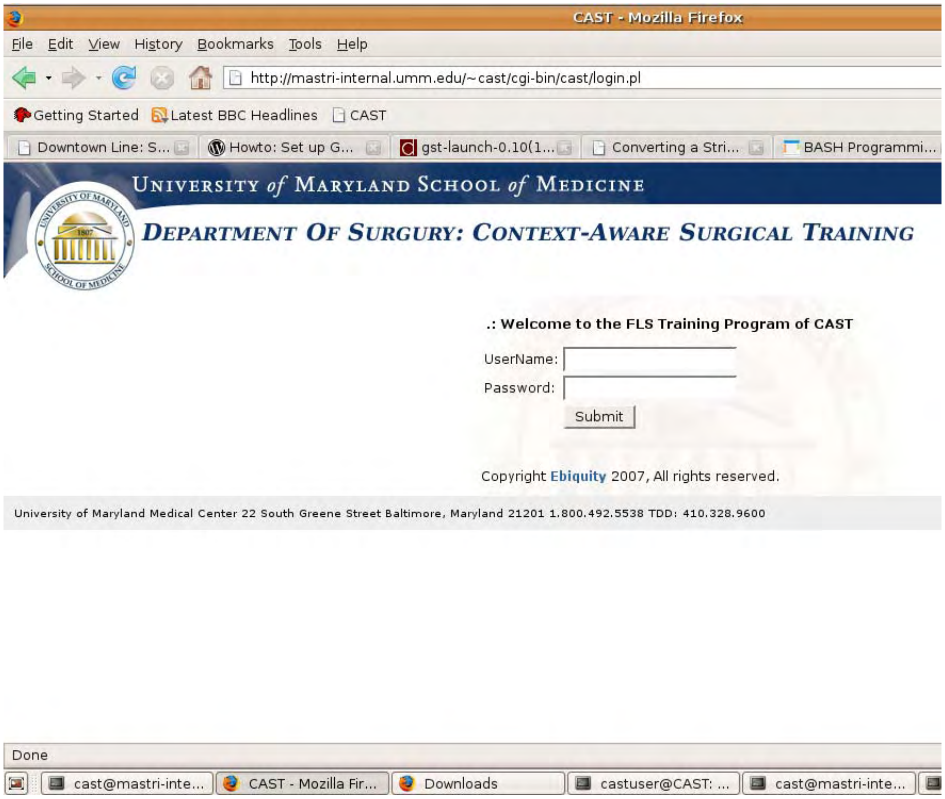
We have now developed the system to capture internal video feeds and metrics from the following simulators

- Promis
- Stryker
- Laproscopic VR simulator

We use external s-video frame grabbers to capture the simulator internal video feeds. These feeds are synchronized with the external view from the N800 and stored on the video server. Thus, the instructor now has access to both the internal and external feeds during review, and consequently they can provide better feedback. Currently our system uses email to send back feedback.



Web Interface:



UNIVERSITY *of* MARYLAND SCHOOL *of* MEDICINE

DEPARTMENT OF SURGURY: CONTEXT-AWARE SURGICAL TRAINING

.: Welcome to the FLS Training Program of CAST

UserName:

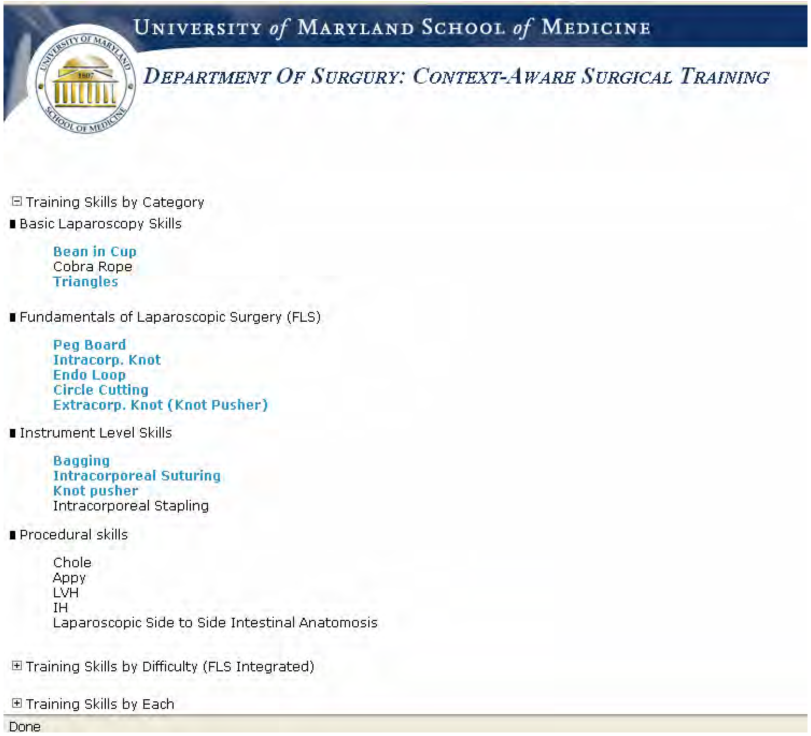
Password:

Submit

Copyright Ebiquity 2007, All rights reserved.

University of Maryland Medical Center 22 South Greene Street Baltimore, Maryland 21201 1.800.492.5538 TDD: 410.328.9600

We have integrated the student database into our web-based curriculum management system. The student database contains all the current residents and one guest account that can be used for testing. If a student wants to view the training videos, he or she will first need to log in using their SMail Userid. When they pass the authentication, a categorized web structure will be displayed, and they can choose to sort the tasks by category, by difficulty (FLS integrated), or by each (Basic, Instrument, Procedural Skills, and FLS), which is shown in the following screenshot. This structure was developed in consultation with the MASTRI team, particularly Ivan George and Ethan Hagan.



University *of* Maryland School *of* Medicine

Department Of Surgury: Context-Aware Surgical Training

⊟ Training Skills by Category

- Basic Laparoscopy Skills

  Bean in Cup
  Cobra Rope
  Triangles

- Fundamentals of Laparoscopic Surgery (FLS)

  Peg Board
  Intracorp. Knot
  Endo Loop
  Circle Cutting
  Extracorp. Knot (Knot Pusher)

- Instrument Level Skills

  Bagging
  Intracorporeal Suturing
  Knot pusher
  Intracorporeal Stapling

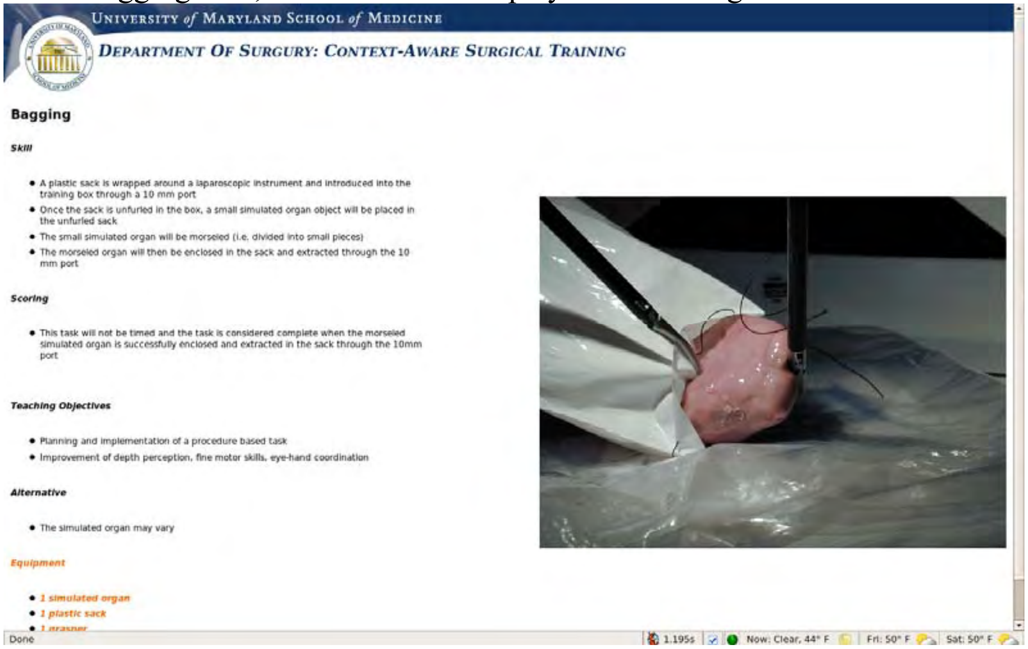
- Procedural skills

  Chole
  Appy
  LVH
  IH
  Laparoscopic Side to Side Intestinal Anatomosis

⊞ Training Skills by Difficulty (FLS Integrated)

⊞ Training Skills by Each

Then the student can pick any training video they want to view. Suppose the student want to view the bagging skill, and then it will display the following screen.



University *of* Maryland School *of* Medicine

Department Of Surgury: Context-Aware Surgical Training

Bagging

Skill

- A plastic sack is wrapped around a laparoscopic instrument and introduced into the training box through a 10 mm port
- Once the sack is unfurled in the box, a small simulated organ object will be placed in the unfurled sack
- The small simulated organ will be morseled (i.e. divided into small pieces)
- The morseled organ will then be enclosed in the sack and extracted through the 10 mm port

Scoring

- This task will not be timed and the task is considered complete when the morseled simulated organ is successfully enclosed and extracted in the sack through the 10mm port

Teaching Objectives

- Planning and implementation of a procedure based task
- Improvement of depth perception, fine motor skills, eye-hand coordination

Alternative

- The simulated organ may vary

Equipment

- *1 simulated organ*
- *1 plastic sack*
- *1 grasper*

As for the instructor interface, the instructor first needs to pass authentication to access the student training records. Then, they can pick the student name that they want to view from a drag-down list that contains all the residents. The appropriate student record will pop up in the next page, which contains the following information: the chapters that the student has checked out, the student training history (simulator type, start time, end time, internal video record, external video record), and the instructor can provide their feedback for every training record of this student via email.

Informatics subgroup 4. Operating Room Clutter (ORC)

The project team has worked on the use of advanced video technology to support coordination in operating rooms. Our activities were in four areas. All publications referred to may be found in our website: http://hfrp.umaryland.edu. For full length journal articles, PDF files may be downloaded. For others, abstracts are available. In all, we published 8 full-length peer reviewed journal articles, 2 full-length peer reviewed proceeding articles, and 8 conference abstracts. The references below can provide further details.

A. Models of decision making for operating room management.

We reviewed literature and developed a synthesis report on the state of the art of decisions on the day of surgery. Furthermore, we developed models for decision support systems for operating room management. The activities in this area were reported in the following publication:

1. Dexter F, Xiao Y, Dow AJ, Strader MM, Ho D, Wachtel RE. Coordination of Appointments for Anesthesia Care Outside of Operating Rooms Using an Enterprise Wide Scheduling System. *Anesthesia and Analgesia. 105:1701-1710. 2007*

2. Xiao Y, Strader M, Hu P, Wasei M, Wieringa P. Visualization Techniques for Collaborative Trajectory Management . *ACM Conference on Human Factors in Computing Systems, pp.1547 - 1552. 2006*

3. Xiao Y, Wasei M, Hu P, Wieringa P, Dexter F. Dynamic Management in Perioperative Processes: A Modeling and Visualization Paradigm. *12th IFAC Symposium on Information Control Problems in Manufacturing. (3)647-52. 2006*

4. Dutton R, Ho D, Hu P, Mackenzie CF, Xiao Y. Decision Making by Operating Room Managers: The Burden of Changes. *Anesthesiology, 103:A1175. 2005*

5. Dexter F, Epstein RH, Traub RD, & Xiao Y. Making Management Decisions on the Day of Surgery Based on Operating Room Efficiency and Patient Waiting Times. *Anesthesiology, 101(6):1444-1453. 2004*

B. Operating room multimedia system design and methodology.

We developed technology, primarily based on algorithms of video processing and biosignal processing, to display status of operating rooms. The displays are to increase situational awareness. The technological advances made by our group were reported in the following publications:

6. Xiao Y, Schimpff S, Mackenzie CF, Merrell R, Entin E, Voigt R, Jarrell B. Video Technology to Advance Safety in the Operating Room and Perioperative Environment. *Surgical Innovation. 14(1): 52-61. 2007*

7. Hu P, Xiao Y, Ho D, Mackenzie CF, Hu H, Voigt R, Martz D. Advanced Visualization Platform for Surgical Operating Room Coordination: Distributed Video Board System**.** *Surgical Innovation. 13(2):129-135. 2006*

8. Hu P, Seagull FJ, Mackenzie CF, Seebode S, Brooks T, XiaoY. Techniques for Ensuring Privacy in Real-Time and Retrospective Use of Video. *Telemedicine and e-Health, 12(2): 204, T1E1. 2006*

9. Xiao Y, Hu P, Hu H, Ho D, Dexter F, Mackenzie CF, Seagull FJ, Dutton D. An algorithm for processing vital sign monitoring data to remotely identify operating room occupancy in real-time. *Anesthesia & Analgesia,(101)3:823-829 . 2005*

10. Hu PF, Burlbaugh M, Xiao Y, Mackenzie CF, Voigt R, Brooks T, Fraser L, Connolly MR, Herring T. Video Infrastructure and Application Design Methods for an OR of the Future. *Telemedicine and e-Health. 11(2), 211, T3C2. 2005*

11. Hu PF, Hu H, Seagull JF, Mackenzie CF, Voigt R, Martz D, Dutton R, Xiao Y. Distributed Video Board: Advanced Telecommunication System for Opearation Room Coordination**.** *Telemedicine and e-Health. 11(2), 248, P28. 2005*

12. Hu PF, Xiao Y, Mackenzie CF, Seagull FJ, Brooks T, LaMonte MP, & Gagliano D. Many to One to Many Telemedicine Architecture and Applications. *Telemedicine Journal and e-Health. 10(Supplement 1), S-39. 2004*

C. Survey and descriptive studies of operating room management, with and without the support of advanced video technology.

In conjunction with technology development, we conducted observational and survey studies of operating room management. These studies and associated results were in the following publications:

13. Seagull FJ, Xiao Y, & Plasters C. Information Accuracy and Sampling Effort: A Field Study of Surgical Scheduling Coordination**.** *IEEE Transactions on Systems, Man, and Cybernetics, Part A:Systems and Humans. 24(6), 764-771. 2004*

14. Dutton R, Hu PF, Mackenzie CF, Seebode S, Xiao Y. A Continuous Video Buffering System for Recording Unscheduled Medical Procedures. *Anesthesiology, 103:A1241. 2005*

15. Gilbert TB, Hu PF, Martz DG, Jacobs J, Xiao Y**.** Utilization of Status Monitoring Video for OR Management. *Anesthesiology, 103:A1263. 2005*

16. Dutton R, Hu P, Seagull FJ, Scalea T, Xiao Y, . Video for Operating Room Coordination: Will the Staff Accept It?. *Anesthesiology: 101: A1389. 2004*

D. Technology evaluation.
We conducted evaluation studies of the technology deployed. The primary focus was on user acceptance and usage patterns. The focus was chosen because the current science of operating room management has concluded that improvement of decision making on the day of surgery will lead to improvement in intangible outcomes, such as situation awareness, and will unlikely lead to improvement in operating room throughput (e.g., volumes and economic returns). Our work was reported in the following publication.

17. Xiao Y, Dexter F, Hu FP, Dutton R. Usage of Distributed Displays of Operating Room Video when Real-Time Occupancy Status was Available . Anesthesia and Analgesia 2008; 106(2):554-560. 2008

18. Kim Y-J, Xiao Y, Hu P, Dutton RP. Staff Acceptance of Video Monitoring for Coordination: A Video System to Support Perioperative Situation Awareness. *Journal of Clinical Nursing (accepted). 2007*

Informatics subgroup 5. Improving Perioperative Communications (IPC)

Reassignment of responsibilities:

In order to streamline work for this task we have reorganized the IPC team to include the following (no cost) individuals: Ramon Konewko, MSN: Perioperative scheduling/Effectiveness coordinator will give a greater perspective of the posting issues; Tim Clinton: (ad hoc.) IT Sr. Administrator/Manager; Barbara Galobolski: Communications Specialist; Max Warnock: IT Support; Rommel Moon: Communications Specialist. Previous team members remain engaged; Ivan George, Paul Nagy. PhD and Wendy Bernstein, MD.

Data Analysis

The team has actively performed a re-review/study of the performance metrics available from various Perioperative data stores. In particular, near past analysis has been benefited by work performed via the OGA team.

Problem Statement Development:

The team has sought to find an appropriate "question" with which to focus this effort. In particular, there was a need to tie a performance metric (perioperative workflow related) to the IPC task. During the end of this quarter, with the inclusion of new team member, the team was able to resolve this issue.

Background:

In the UMMS OR the Cardiac Surgery Service utilizes a common communications point (a "cardiac phone line") that in a sense is used to acquire information and provide that information to any team member who calls the line to acquire information. The cardiac phone line has been scripted and is actively in use through a voice mail system. It can only be altered by dedicated personnel with password capability. The script involves the following standardized information: Identification of individual providing information, the Date of surgery, the Total number of cases, and OR location, patient name, case order, medical record number, age, surgeon, anesthesiologist and procedure. Evening schedule updates have been made possible through a second phone line option.

Problem Statement:

After some effort, we can now move to track updates on the phone line and correlate these updates with OR start delays. Thus, we refined the IPC question to Does more accurate information as evidenced by updates on the phone line, ie improved communication, result in fewer problems in the morning with cardiac surgical cases starting on time- are instruments better prepared for the procedure, are operating rooms better equipped for the appropriate case, are the correct pick lists utilized for the correct surgeon, is there less of a transport delay because the patient's hospital location has been identified? The question contains reference to some of the delay codes that are currently utilized by the Operating room tracking system and reported for glitch analysis.

With the assistance of the communications personnel we will reconfigure the cardiac phone line so that we can actually track the phone calls made to the phone line. This will enable us to: Determine key personnel who are utilizing the phone line, Determine groups of personnel utilizing the phone line (i.e. nursing, anesthesia, perfusion), Determine which groups are not utilizing phone line information (i.e. anesthesia techs), Determine whether there is a time variable; is there a better time to call for updates? Should updates be made at predetermined times or should they be more dynamic?

We hypothesize that information gained from increased communication improves OR efficiency. If this is the case we can them move to see if more real-time enabling technologies might be deployed to other services within the UM ORs and perhaps other ORs "everywhere".

After a slow start, the IPC project is moving forward and has moved to identify and utilize new technologies (Cell, WiFi, IM, Web fusion technologies) being developed in the UM Radiology department. We hope to expand this simple phone line to a form of communication that is more mobile, accurate, up-to-date, and shares a common lexicon.

Time line adjustments

In order to better portray the reenergized IPC plan, the following revised timeline is set: NOTE: Lined through items have been completed. There is no change in the SOW.

Revised TIMELINE: Improving Communication for Cardiac Surgical Cases

1. Pre planning
2. Determine necessary information to be conveyed and appropriate individuals
3. Develop scripts and phone line
4. Review technological components of accessing phone logs and usage information
5. Map process current
6. Analyze process
7. Develop process to access all callers to cardiac phone lines

a. Determine their respective services and information retrieved
b. Determine effectiveness of phone line in achieving appropriate information
c. Determine ancillary services that are not routinely accessing phone line information
d. Correlate access issues with
i. Surgical delays in first case starts
ii. Staffing problems with misinformation on phone line
iii. Staff satisfaction surveys pre and post implementation of phone line changes

8. Review, Develop and Deploy Technology and Procedural Systems.

a. Survey of available options ongoing

Specific interest in UM Radiology Critical Reporting system!

b. Determination of key stake holders
c. Determine necessary information to convey/ initiate prompt and alert system
d. Design wireless system to meet specs derived from summative analysis.
e. Deploy enabling technological system
f. Analyze benefits.
g. Publish findings.

Timeline shift: We are now about to start step 7 and we anticipate a slight right shift for the remaining schedule. Line items stricken out represent those milestones completed

B. Simulation

This report introduces the basics of the Maryland Virtual Patient simulation approach, discusses its place on the map of intelligent systems in clinical medicine and describes the project's status and research and development activity presently under way. The complete report appears in Appendix B.

Simulation: The Maryland Virtual Patient

We present here a simplified description of the MVP simulation, interaction and tutoring system. A virtual patient instance is launched and starts its simulated life, with one or more diseases progressing. When the virtual patient develops a certain level of symptoms, it presents to the attending physician, the system's user.[1] The user can carry out, in an order of his or her choice, a range of actions: interview the patient, order diagnostic tests, order treatments, and schedule the patient for follow-up visits. The patient can also automatically initiate follow-up visits if its symptoms reach a certain level before a scheduled follow-up. This patient-physician interaction can continue as long as the patient "lives."

As of the time of writing, the implemented MVP system includes a realization of all of the above functionalities, though a number of means of realization are temporary placeholders for more sophisticated solutions, currently under development.[2] The most obvious of the temporary solutions is the use of menu-based patient-user interaction instead of natural language interaction. While this compromise is somewhat unnatural for our group, which has spent the past 20 years working on knowledge-based NLP, it has proved useful in permitting us to focus attention on the non-trivial core modeling and simulation issues that form the backbone of the MVP system.

MVP currently covers six esophageal diseases pertinent to clinical medicine: achalasia, gastroesophageal reflux disease (GERD), laryngopharyngeal extraesophageal reflux disease (LERD), LERD-GERD (a combination of LERD and GERD), scleroderma esophagus and Zenker's diverticulum.[3]

At the beginning of a simulation session, the system presents the user with a virtual patient about whose diagnosis he initially has no knowledge. The user then attempts to manage the patient by conducting office interviews, ordering diagnostic tests and prescribing treatments.

Answers to user questions and results of tests are stored in the user's copy of the patient profile, represented as a patient chart. At the beginning of the session, the chart is empty and the user's cognitive model of the patient is generic – it is just a model of the generalized human. The process of diagnosis results in a gradual modification of the user's copy of the patient's profile so that in the case of successful diagnosis, it closely resembles the actual physiological model of the patient, at least, with respect to the properties relevant to the patient's complaint. A good analog to this process of gradual uncovering of the user profile is the game of Battleship, where the players gradually determine the positions of their opponent's ships on a grid.

At any point during the management of the patient, the user may prescribe treatments.[4] In other words, the system allows the user not only to issue queries but also to intervene in the simulation, changing property values within the patient. Any single change can induce other changes – that is, the operation of an agent can at any time activate the operation of another agent.

Simulation: Utility

The MVP project can be viewed as just one of a number of applications in the area of intelligent clinical systems. The latter, in turn, can be viewed as one of the possible domains in which one can apply modeling teams of intelligent agents featuring a combination of physical system simulation and cognitive processing.

So, in the most general terms, our work can be viewed as devoted to creating working models of societies of artificial intelligent agents that share a simulated "world" of an application domain with humans in order to jointly perform cognitive tasks that have until now been performed exclusively by humans. Sample applications of such models include:

- a team of medical professionals diagnosing and treating a patient (with humans playing the role of either a physician or a patient)
- a team of intelligence or business analysts collecting information, reasoning about it and generating analyses or recommendations (with humans playing the role of team leader)
- a team of engineers designing or operating a physical plant (with humans playing the role of team leader)
- a learning environment (where humans play the role of students).

As can be seen, this work is at the confluence of several lines of research – cognitive modeling, ontological engineering, reasoning systems, multi-agent systems, simulation and natural language processing.

Simulation: Accomplishments

In Year 4 of the project, our team has delivered two new versions of the Maryland Virtual Patient Environment. The realism of the simulation has been enhanced by including coverage of "unexpected" interventions; allowing discontinued treatments; allowing new diseases to develop due to side effects of treatments. The user interface has been redesigned. A new agent-based architecture has been developed to support enhanced cognitive capabilities of the virtual patient and the intelligent tutor, including language capabilities. In the area of language processing, a dialog processing model was developed. Work has continued on improving the language understanding capabilities, centrally including treatment of referring expressions. Enhancement of static knowledge resources, the ontology and the lexicon, has been ongoing. Work on extending the coverage of diseases has been ongoing: a further improvement of the model of GERD is under way, as is the modeling of cardiovascular diseases. A totally reworked system version, with dialog support, is planned for release in June 2008. Work has also been ongoing on improving and extending the set of development tools – the DEKADE demonstration, evaluation and knowledge acquisition environment supporting natural language work has been revamped; the interface for creating instances of virtual patients has also been enhanced; a web-based environment for supporting internal documentation has been installed. Finally, we have written, submitted, published or delivered 6 conference and journal papers.

C. Smart Image

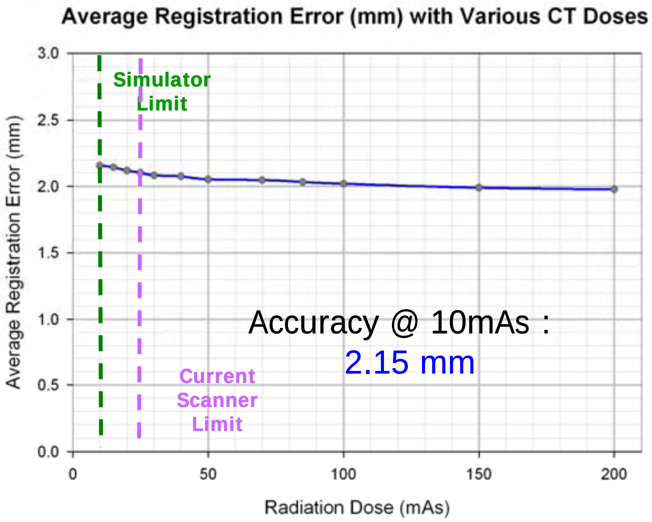
C.1. Smart Image: CT guided imaging

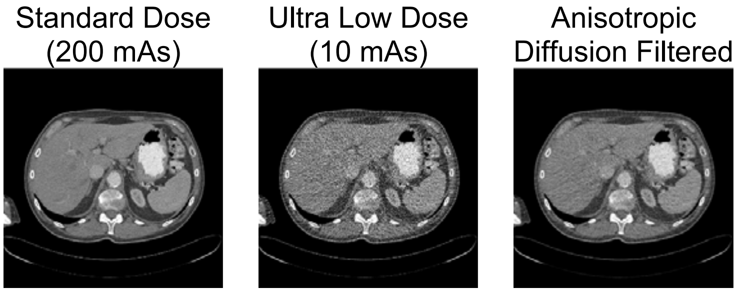
The overall objective of this project is to demonstrate the technical feasibility of live augmented reality (AR), which is the fusion of instantaneous computed tomography (CT)-generated views with laparoscopic views. The advantage of live AR is that internal structures that are absent in laparoscopic views can be visualized. Being able to see underlying vessels and other internal structures before making a dissection has been a longstanding need of the minimally invasive surgeons.

Although the proposed use of continuous CT is advantageous to creating renderings of the internal structures with their orientations updated at real-time rates, it is also imperative that methods be created to reduce the radiation exposure to the patient and the surgeon alike through the use of CT. Our first three objectives address the radiation dose problem while providing a means to visualize the vasculature throughout the surgery with a single administration of the CT contrast agent. The fourth objective is devoted to creating spatially and temporally synchronized AR views. The last objective will integrate all the individual technology solutions together when those are developed.

Objective 1: Dose reduction strategy: Registration between High dose-Low dose CT

The ultimate goal of this objective is to be able to track tissue motion through use of intra-operative low-dose CT scans and their deformable registration with peri-operative, diagnostic-quality CT scan. We have demonstrated the feasibility of intensity-based deformable registration between low-dose CT and regular-dose CT [1]. This preliminary study preformed registration between standard-dose CT (representing a pre-operative image) and ultra low-dose CT (representing intra-operative image). Even at 10 mAs, the smallest dose achievable, the registration accuracy achieved was comparable to that achieved at the standard dose. These results (see figure below) demonstrate the potential for ten- to twenty fold reduction in radiation dose with the use of low-dose CT. We are currently working on extending this study using human data.





Pre-processing of Low-dose CT

One of the challenges of working with low-dose CT is the poor quality (signal-to-noise ratio) of these images. We have developed image preprocessing techniques to eliminate this noise and the make these image more suitable for intensity-based deformable. image registration. The results of these preprocessing techniques are shown in the following figure. For on-line use, however, these pre-processing steps must be accelerated such that the pre-processing operations can be performed in real-time. We have developed and reported [2] an FPGA-based implementation that provides over an order of magnitude speedup and is capable of performing this preprocessing in few milliseconds.

In addition to the developments mentioned above, we have also attempted to validate the feasibility of the above approach using animal datasets. This was achieved by using the low-dose and regular-dose CT data collected during the animal experiment. However, due to a localized deformation created using an ad-hoc method (syringe-pump), the deformable registration algorithm did not produce expected results. We investigated this issue and devised strategies to introduce a controlled and reproducible deformation. For further testing of these ideas we performed an experiment using a chicken, which allowed us to introduce reproducible deformations in a controlled fashion. These deformations were then track through the use of intra-operative low-dose CT. The low-dose CT data collected during this chicken experiment was processed. We registered this data with pre-operative regular dose scans and the results indicated visually acceptable quality of registration. This demonstrates the preliminary feasibility of deformable registration with low-dose CT in animal datasets.

Objective 2: Dose reduction strategy: Iterative reconstruction

Iterative reconstruction techniques are known to give better quality image reconstructions in the presence of noise. Hence these techniques are better suited for reconstruction of low dose images. Over the past year we have implemented the maximum Likelihood Expectation Maximization technique for reconstruction of images.

The algorithm has given better quality images for simulated low dose as well as normal dose data sets. However, since the MLEM is an iterative technique with typical reconstruction involving about 40 iterations, and each iteration involving one forward and one back projection, the algorithm is not very attractive for real-time processing over a uniprocessor.

To enable faster implementation of this computationally intensive algorithm, a parallel version of the same has also been implemented which can be executed on a cluster of multiple processors. The cluster based approach gives almost linear speed-ups as compared to a uniprocessor implementation. Table 2.1 below indicates the speed-ups achieved for a representative data set.

Xeon,3.6GHz,1.5GB RAM	1 core of a Quad-core Xeon node, 3.2GHz, 4GB RAM	32 cores on 8 Quad-core Xeon nodes, 3.2 GHz, 4GB RAM
180 sec	143 sec	5.54 sec

*Table 2.1 Iteration time for a 256*256 CT slice with 367 detectors and 360 projections*

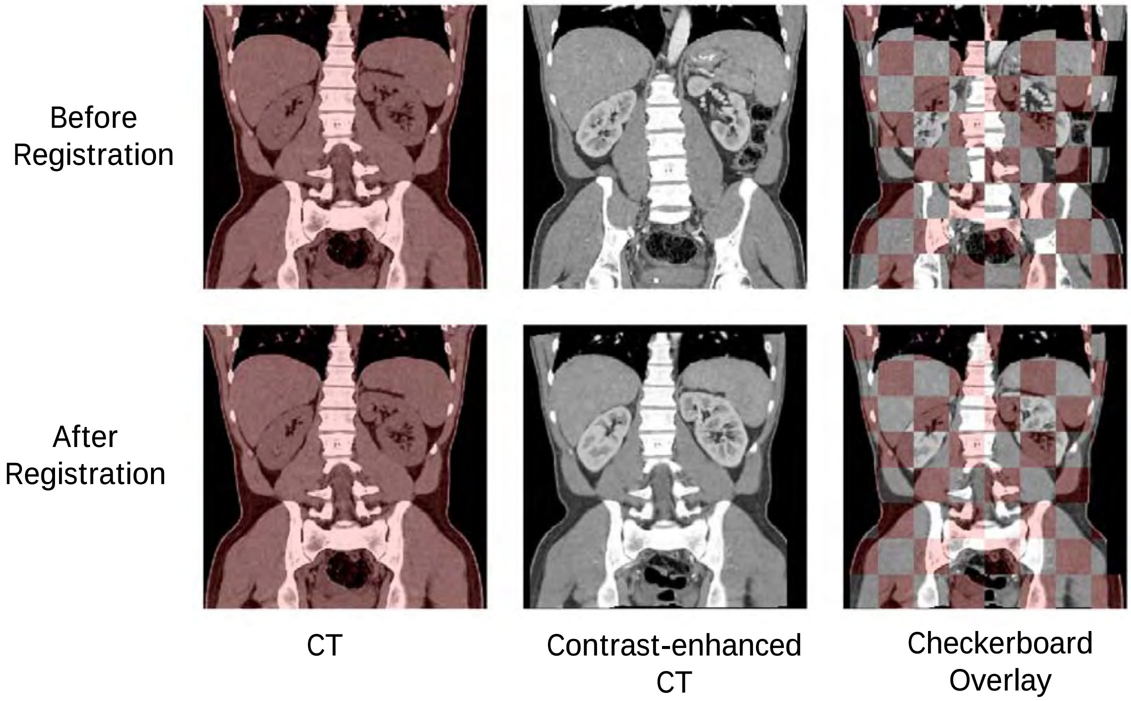
The reconstructed image quality was the same irrespective of the amount of parallelization. To further speed-up the rate of convergence, an Ordered Subset version of the Maximum likelihood algorithm was also implemented. This version was also parallelized and can now be run on a cluster of multiple processors. The speed-up achieved was linear to the number of subsets used in the algorithm.

The implementation of the MLEM as well as OSEM algorithms is now being optimized to be more memory efficient to enable quick reconstruction of real data sets. To this end we are currently working on a low memory imprint version of the OSEM algorithm as well as a GPU based implementation of the same algorithm.

Objective 3: High-speed implementation of non-rigid registration

Deformable registration between intra-operative images and peri-operative images is a fundamental need in image-guided procedures. This registration will allow the fusion of complimentary information such as spatial information and vasculature information from pre- and peri-operative images respectively. Computational complexity of deformable registration, however, has prevented its use in clinical applications. This objective attempts to address this problem through use of hardware-acceleration.

We presented the initial architectural design for accelerated deformable image registration [1]. This architecture is capable of calculating mutual information, a compute intensive step in intensity-based image registration, around 40-times faster than a software-based implementation can reduce the execution time of deformable registration from hours to minutes. The detailed design of this architecture was later published in IEEE - Transactions on Biomedical Circuits and Systems [2]. This architecture has been validated for CT-CT registration and will allow deformable registration in a matter of few minutes. The validation was performed using 5 CT-contrast-enhanced CT image pairs and the results of registration are presented in the following figure. Also, the accompanying table compares the execution times of this architecture against a software implementation and reports the achieved speedup.



As a first step towards implementation of deformable image registration, we completed the hardware implementation of mutual information (MI)-based rigid registration. This implementation has been fully tested and rigorously validated using clinical and artificially deformed datasets. The accuracy achievable through this implementation is comparable to that achieved by a software implementation. This implementation is capable of providing 40-fold speedup in MI calculation and can achieve rigid registration is 50 seconds. Currently, we are working on optimizing this implementation for accuracy and hardware resources and a manuscript based on this work has been submitted to IEEE Conference on Field-Programmable Custom Computing Machines [3]. Once this optimization is complete, we will finalize and test the hardware implementation for deformable registration.

Comparison of execution time for deformable registration

Image Modality	**Software Implementation**	**Hardware Implementation**	**Speedup**
CT-Contrast CT	11520 s	371 s	31

Objective 4: Tracking and visualization

We have developed the mechanism to track the laparoscope and other tools optically. Image 4.1 shows our experimental setup for tracking of operative tools.

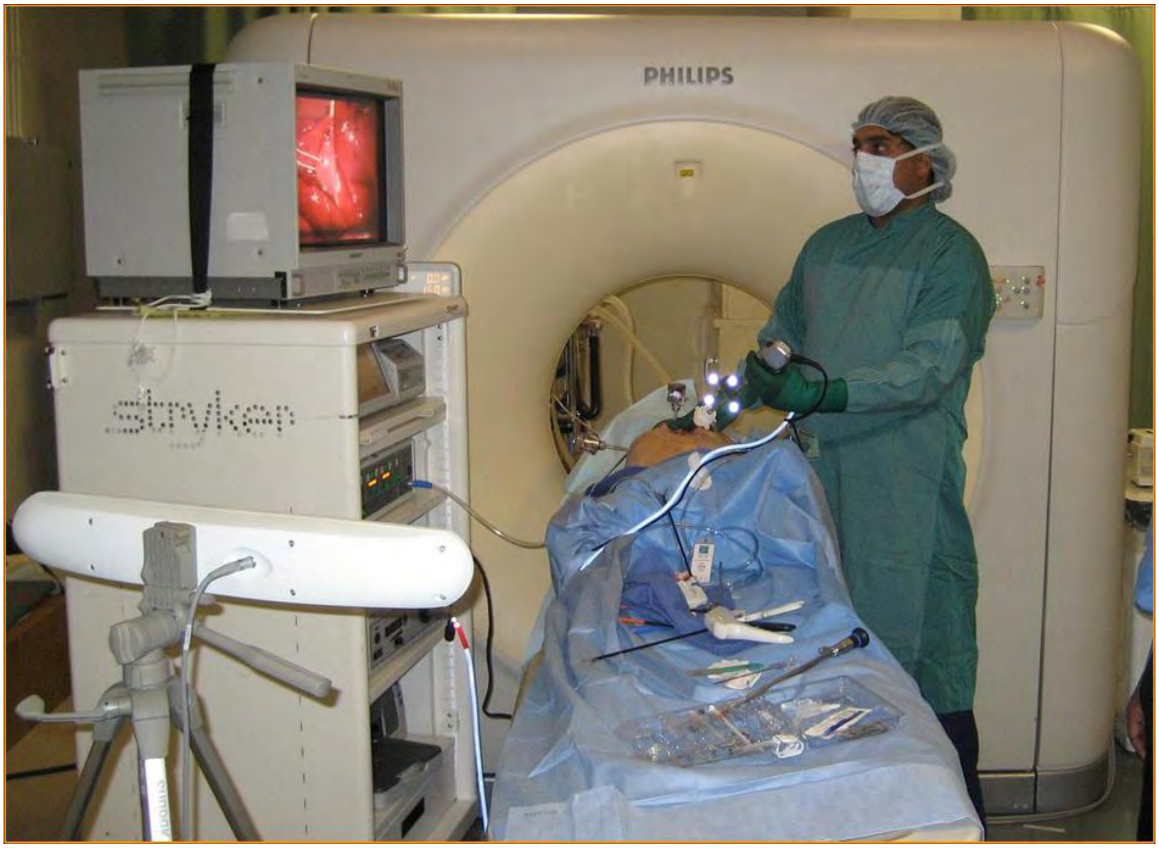


Image 4.1: Experimental setup for optical tracking of intra-operative tools.

2 experiments were conducted over the duration of the last year to collect data and to improve the accuracy and the calibration methods used in the data tracking and collection methods. Each of the experiments has yielded progressively improved results. Some of the major improvements in the tracking and visualization methods due to the experiments are as follows:

1. We have moved to automatic volume rendering of the CT data for fast automatic 3-D volume generation as against the previous segmentation techniques.

2. Camera calibration has been integrated as a crucial step in the calibration procedure to account for the optical characteristics of the endoscope.

Image 4.2: Camera calibration plate with optical trackers

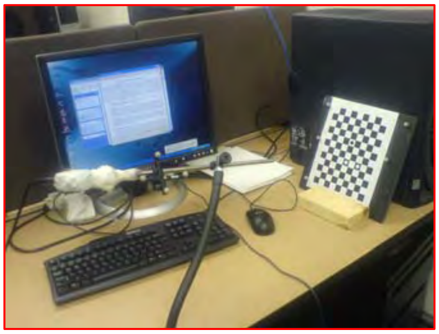


Image 4.3: Camera calibration setup

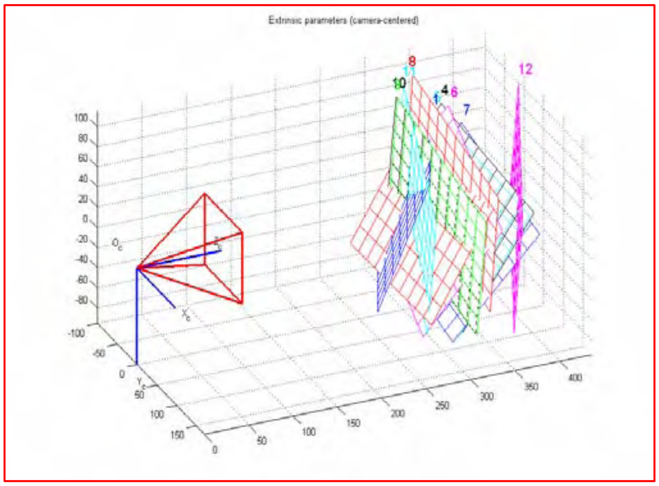


Image 4.4: Various positions of the calibration plate for camera calibration

3. Most of the steps in the tracking and visualization procedure have been completely automated to achieve fast and efficient registration of the CT reconstructed images with the optical images.

Though the experimental results have been improving, we still are facing some issues with the spatial and temporal registration of the optical images from the laparoscope and the 3-D volume rendered images from the CT scans. We are therefore working on isolating the unknown parameters in the tracking and visualization setup and devising improvements in our calibration techniques to overcome the problems.

Challenges

While our current efforts will prove the feasibility of live AR, implementing a real-time system for routine clinical use will require meeting a few additional technical challenges. First of all, the current generation CT scanners remain ill-suited for the task. Further development of the CT technology will be needed to improve the speed of both scanning and reconstruction to make CT real-time. We are developing a relationship with the manufacturer of our CT scanners (Philips) to address this challenge.

The use of multi-vendor devices is a challenge in creating spatially and temporally registered live AR display. Different devices exhibit different latencies, which are being estimated experimentally currently. Knowing those, or perhaps eliminating those, will improve the accuracy of live AR.

Finally, image reconstruction and registration are computationally intensive tasks. Although our present will show considerable acceleration of both, further acceleration will be needed to create a real-time system. We believe our current work will provide the necessary impetus to form necessary partnerships and take on the described challenges in a follow-on phase.

References:

O. Dandekar, K. Siddiqui, V. Walimbe, and R. Shekhar, "Image registration accuracy with low-dose CT: how low can we go?," in 3rd IEEE International Symposium on Biomedical Imaging: Nano to Macro, 2006, pp. 502-505.

O. Dandekar, C. Castro-Pareja, and R. Shekhar, "FPGA-based real-time 3D image preprocessing for image-guided medical interventions," Journal of Real-Time Image Processing, vol. 1(4), pp. 285-301, 2007.

A. S. Shetye and R. Shekhar, "A statistical approach to high quality CT reconstruction at low radiation doses for real-time guidance and navigation," Proc. SPIE Med. Imaging, 2007.

O. Dandekar, V. Walimbe, and R. Shekhar, "Hardware Implementation of Hierarchical Volume Subdivision-based Elastic Registration" in 28th Annual International Conference of the IEEE: Engineering in Medicine and Biology Society, 2006, pp. 1425-1428.

O. Dandekar and R. Shekhar, "FPGA-accelerated Deformable Registration for Improved Target-delineation During CT-guided Interventions," IEEE Transactions on Biomedical Circuits and Systems, vol. 1(2), pp. 116-127, 2007.

O. Dandekar, W. Plishker, S. Bhattacharyya, and R. Shekahr, "Multiobjective Optimization of FPGA-Based Medical Image Registration" IEEE Symposium on Field-Programmable Custom Computing Machines, Under Review, 2008.

R. Shekhar, O. Dandekar, S. Kavic, I. George, R. Mezrich, and A. Park, "Development of continuous CT-guided minimally invasive surgery," Multimedia Meets Virtual Reality (MMVR), 2007.

R. Shekhar, O. Dandekar, S. Kavic, I. George, R. Mezrich, and A. Park, "Development of continuous CT-guided minimally invasive surgery," Proc SPIE, Medical Imaging 2007.

C.2. Smart Image: Image Pipeline

Multimodal Registration Development and Evaluation (Intra-operative)

During the past year, we have developed and evaluated a technique to address the classic multimodal registration problem. The goal of our technique is to allow the integration of multiple laparoscopic views into a unified (3D, wide field of view) image. The technique has the following characteristics:

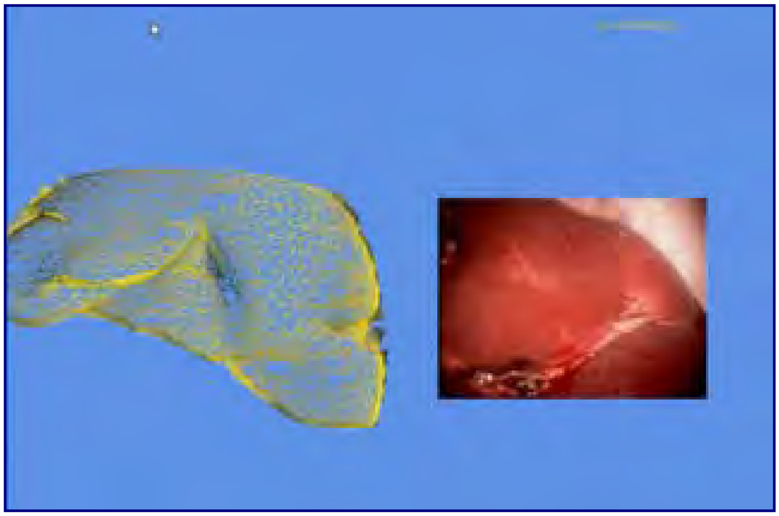
a. It does not rely on a tracked camera (traditional approach). This is important because it is difficult to get accurate camera parameters using available tracking techniques. As a result, our evaluation of the traditional method produces obvious shifts between the target (desired) and actual (obtained) images.

b. It can deal with deformable objects. This is important because of the deformations introduced to objects during surgical procedures.

c. It changes the difficult 2D-3D registration problem into a 2D-2D registration problem which is better understood.

Our approach begins from a single interactively aligned image/3D model pair. It involves an incremental global-optimization algorithm to automatically register additional nearby images. We demonstrated a registered data set during the ORF 2007 conference. This example is presented below.

.



We have more formally evaluated the approach by testing the simplified case of a sphere surface. Compared with ground truth, the error was calculated by determining the average distance of texture coordinates per vertex (482 vertices, 960 faces, and 81 feature points, mean error per vertex < .0045).

We next turned to a test case for our registration technique that was considerably more complex than a sphere – a human face and a set of teeth. The results of this study were described in a paper, titled "Feature-based Texture Mapping from Video Sequences," that was accepted as a poster presentation at the Symposium on Interactive 3D graphics and Games (I3D 2008). The full paper is attached to this report.

Example of feature-based texture mapping applied to a face model to create a 3d image that retains surface detail.

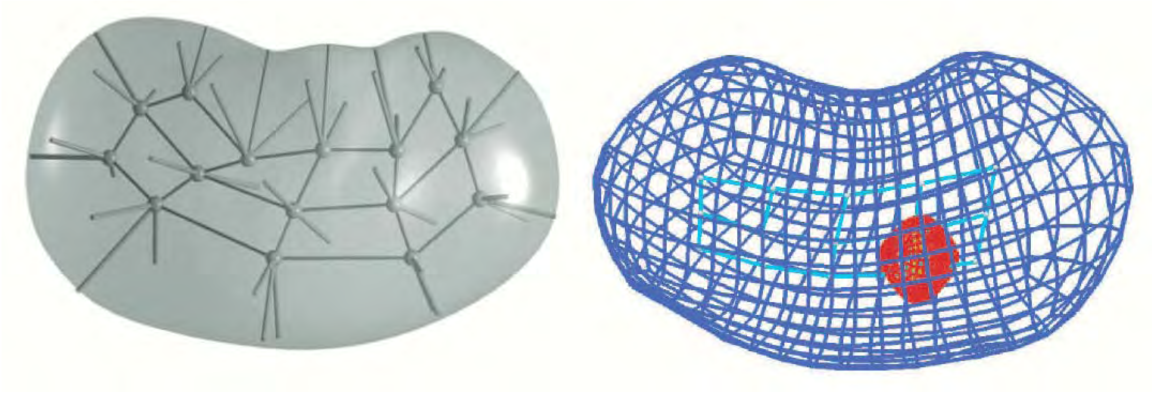
Our most detailed evaluation of our registration technique is currently in progress. We are using as stimuli anatomical models of abdominal organs. 3D data are being obtained by scanning the organs with a Faro Arm (platinum model), a 3-d scanner that uses the position of a fully-articulated arm with a laser-scanner mounted on the end to gather data used to create a 3D model. The surface images of the organs are being acquired using an attached Stryker 1088 endoscope. For comparison purposes, registered output can be generated using the relatively precise motion tracking of the Faro Arm in order to obtain location parameters of the attached endoscope. This will provide us with a baseline condition that utilizes the traditional camera tracking registration methodology with our new technique. In order to make sure that our tracking data are as accurate as possible, a special mount was built to enable a more rigid attachment of the endoscope to the arm.

Initial tests using the Faro arm and attached endoscope have shown promise. Using previous work developed under the REVEAL project, the endoscope's calibration parameters have been extracted from a video sequence of a simple calibration target. These parameters are then combined with the tracking data from the Faro arm to calculate the transformation from the camera to the arm. Software to merge this tracking data with the high-definition video sequence for structure-from-motion 3D reconstruction is also currently in production by Dr. Yang's team

Multimodal Registration Development and Evaluation (Pre- and Intra-operative Data Integration)

In addition to exploring methods for integrating data from multiple intra-operative sources, we have been investigating methods for registering pre- and intra-operative data. This effort has been led by Dr. Qiong Hon, a medical imaging specialist, who is the newest member of the Smart Image team. Our approach is based on the goal of computational efficiency, and to that end we are attempting to register shape models from different intra- and pre-operative modalities into the 2D view rather than attempting to register the images themselves.

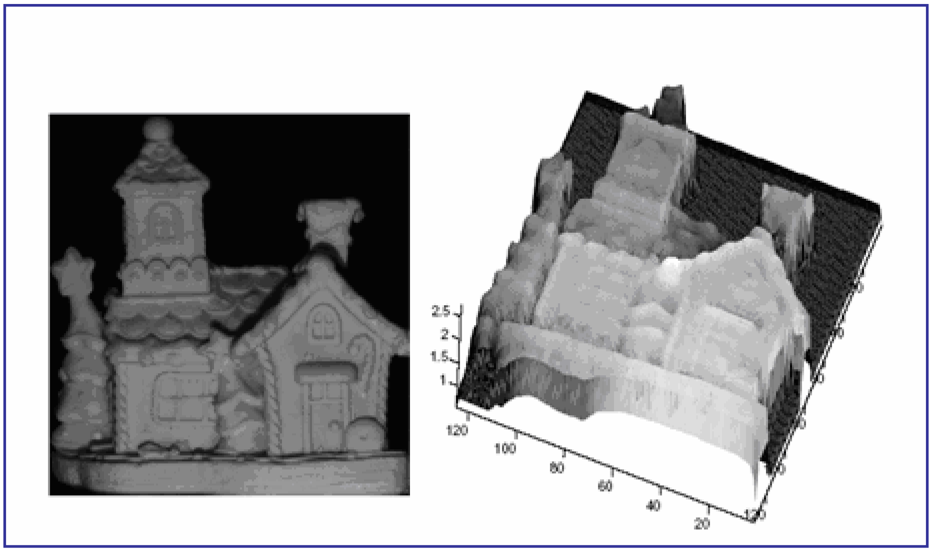
We are using a medially based shape representation called "m-rep" as our shape model. The advantage of an m-rep model lies in its full volumetric parameterization of both the object interior and the adjacent exterior and in its power to capture anatomical shape variations via its non-linear shape parameters. The m-rep has been shown to provide one of the best prostate segmentation results from CT images. Because the m-rep provides a full 3D volumetric shape model, certain manually or automatically extracted features such as landmarks or contours can be highlighted effortlessly. This provides the input for visualization techniques motivated by cognitive ergonomics principles, such as nonphotorealistic rendering, or "decluttering." As described below, we have used the m-rep approach, along with intra-operative texture mapping, to produce a prototype "dual display" visualization that presents anatomical panoramas, detailed camera views, and the relationship between the two simultaneously.



Left: a deformable m-rep shape model; right: a tumor (red) in a kidney

A Novel Intra-operative Depth Acquisition Technique: Light Fall-Off Stereo

The Concept. We have developed a new method for depth acquisition during surgery that takes advantage of the enclosed, light-controlled environment of the surgical site. The technique, dubbed light fall-off stereo (LFS), uses the known properties of light attenuation as a function of distance to infer depth. Our initial results were presented at the International Conference for Computer Vision and Pattern Recognition (CVPR) – one of the premier conferences in the computer vision community. This paper is available at http://vis.uky.edu/~wangl/Research/Publication/2007/Light_fall-off_stereo.pdf. We built a real-time prototype that was presented at the ORF 2007 Conference, examples from which are presented below. While the initial results are quite encouraging, we still need to evaluate its performance with more experiments.

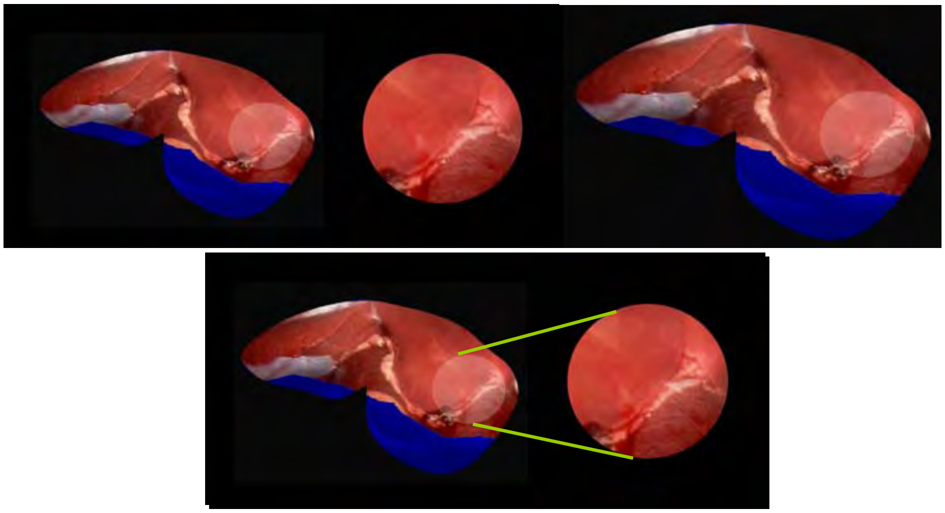


Potential Industry Partnership. Initial results of our work with LFS were successful enough for us to pursue the potential development of a prototype laparoscope that uses our new concept with Stryker Endoscopy. After discussing the project in an initial conference call on 9/27/07 with Stryker marketing representatives, we forwarded relevant research papers and an executive briefing to their engineering department. Stryker showed sufficient interest in our approach to send three representatives, led by Brent Ladd, to visit the UK Center for Visualization and Virtual Environment for a demonstration on 12/5/07. All three Smart Image PIs (Carswell, Seales, and Yang), were present at the meeting. In addition to the demonstrations, we discussed the nature of the light source that would be required for the technique to be effectively incorporated into an endoscope. The Stryker team felt that the technique showed enough promise to invite Dr. Yang to Stryker headquarters for a discussion with its engineers. Stryker offered the first week of March as the time for a visit. Unfortunately, this resulted in a scheduling conflict for the UKY team. We are still negotiating a time to visit.

Smart Imagery Visualization Frameworks: Development and Evaluation

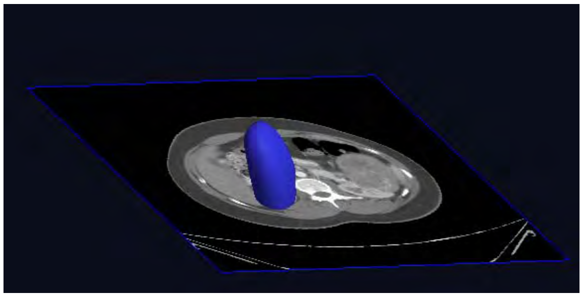
Display concepts. We have continued to pursue the development of visualization techniques that are motivated by cognitive ergonomics principles such as 1) exploitation of redundancy, context, and expectancy, 2) reduction of information access effort, and 3) reduction of memory loads. One such visualization technique is our “dual-view” display. We render the pre-built 3D (m-rep) shape models (described above) along with the original camera view in one of three ways, each method differing in the level of visual integration provided to the user. The shape models are texture-mapped using the panorama textures fused from video sequences of the camera view. In order to link the panorama rendering of the 3D shape models to the original camera, we use the tracked camera pose information to highlight the corresponding portion in the panorama view. In the “separate” dual-view display, the panorama and the camera view are provided in separate windows, with the approximate location of the camera view shown as a circular, highlighted area against the panorama. In the “connected” dual-view display, the panorama and camera views are still in separate windows, but now they are visually tethered by added contours. In the “integrated” dual-view display, the camera view is superimposed in its approximate location on the panorama itself.

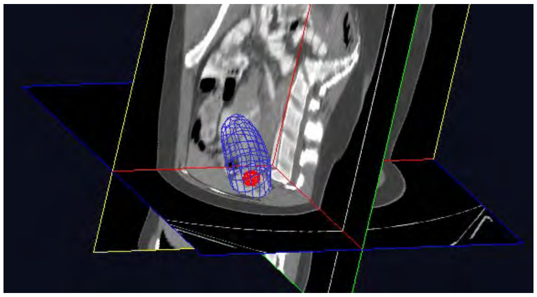
All three dual-view displays show both the concentrated camera view with very limited field of view and the panorama with wide field of view and relate the two views to one another. We have described in more detail the motivation and development of this display in a proposal submitted to CARS 2008 (Principled Display Environment for Computer Augmented Minimally Invasive Surgery, Q. Han, M. Carswell, W. Wang, R. Yang, and B. Seales, Computer-Aided Radiology and Surgery (CARS) 2008.).



Three modes of the dual display environment. Top left: separate mode, top right:: integrated mode, and bottom: linked mode.

In addition to the dual-view displays, we have developed mock-ups of various “hierarchical” displays that highlight different part-whole structures. The “parts” include cross-sections highlighted within the global display, landmarks (e.g., embedded tumors, vascular structures, etc.), strata, metabolic activity gradients, and the like. Each part can be highlighted or removed according to the needs of the surgeon.





Demonstration of the rendering of different parts (information) in our hierarchical displays. Left: a kidney model (in blue) with a tumor sub-model embedded in one axial image slice; right: the same model with the tumor (in red) shown against the axial, sagittal and coronal image slices of the pre-op CT image from which the models are built.

Evaluation. We have developed an evaluation plan that leverages our recent work on situation displays designed for incident commanders of emergency response teams. These commanders, like surgeons, must work with inherently spatial data in safety-critical situations. The assumption behind this work is that different display formats, such as those described above, have differing degrees of "cognitive fit" or compatibility with different tasks. Ideally, the optimal match between tasks and well-designed display alternatives will be self-apparent. That is, the user should be able select the most appropriate display to use for different situations with little additional effort (i.e., without increasing mental workload). To determine whether this is possible, we must design different tasks that are best performed with different displays. That is, we must have normative task-display pairings. These could include, for example, judgments about tumor volume vs. judgments about tumor diameter at the tumor's widest point. A global view should be best for the first task, although a series of cross sections should be better for the second. There should be a large array of "correct" task-display pairings that best support the responses of users. An effectively designed set of rendering options (and these may be conceptualized as the result of user interactions with the "full" global model) will allow rapid and accurate judgments of appropriateness of the various formats by users. So, does Participant X "pull" a cross-section from the global model to solve a problem for which it is the best-suited format?

We have begun development of the experimental scenarios at the same time that Dr. Cindy Lio, our lead usability analyst, is performing a more traditional needs assessment using our preliminary mockups as focal points of semi-structured interviews with surgeons. We will transcribe the interviews and analyze the transcripts to find emergent themes to gain insights about the surgeons' perspectives on the visual displays.

In addition to the mock-ups for our displays described thus far, we have begun development of a more dynamic system for producing prototypes for higher fidelity user studies. Specifically, Dr. Han is developing a simulation program based on the hierarchical display environment. The FARO robot arm is used as an accurate and reliable tracker to control a virtual camera. The virtual camera pose is then used by the simulation program to generate a simulated laparoscopy camera video sequence. In the simulation both synthetic objects and objects from the real world can be used as the targets, and the ground truth of both the object geometry and deformation field is available.

In addition to the performance tests described above, we will be able to utilize a new eye-tracking system to document display utilization at a more detailed level. We are currently validating the eye tracking system with traditional laparoscopic displays, documenting changes in display utilization as training progresses.

The faceLAB eye-tracking system tracks eye and head positions in real-time with a calibrated pair of stereo cameras. Tracking information from the eyes can be overlaid on video streams to accurately show what objects attract a subject's attention. Other useful metrics for cognitive ergonomics include blink rate, eye closure, saccadic movement, and pupil dilation. Together, this information can provide feedback on a subject's stress, mental workload, and fatigue, giving both the psychologists and the software developer an objective measure of the effectiveness of their work to the end user.

Key Research Accomplishments

A. Informatics

Informatics subgroup 1. Perioperative Scheduling Study

- Conducted interviews with groups involved in surgery scheduling process.
- Formulated the purpose of the study.
- Defined the functionality of the congestion evaluation model.

Informatics subgroup 3. Context Aware Surgical Training (CAST)

- Requirements elicitation from the MASTRI Team
- Developed two prototype versions using the spiral development methodology
- Currently the system is under evaluation at the MASTRI center

Informatics subgroup 4. Operating Room Clutter (ORC)

Publications:

Dexter F, Xiao Y, Dow AJ, Strader MM, Ho D, Wachtel RE. Coordination of Appointments for Anesthesia Care Outside of Operating Rooms Using an Enterprise Wide Scheduling System. *Anesthesia and Analgesia. 105:1701-1710. 2007*

Kim Y-J, Xiao Y, Hu P, Dutton RP. **Staff Acceptance of Video Monitoring for Coordination: A Video System to Support Perioperative Situation Awareness**. *Journal of Clinical Nursing (accepted). 2007*

Xiao Y, Schimpff S, Mackenzie CF, Merrell R, Entin E, Voigt R, Jarrell B. **Video Technology to Advance Safety in the Operating Room and Perioperative Environment**. *Surgical Innovation. 14(1): 52-61. 2007*

Xiao Y, Hu P, Moss J, de Winter J, Venekamp D, Mackenzie CF, Seagull FJ, Perkins S. **Opportunities and challenges in improving surgical work flow**. *Cognition, Technology & Work, accepted. 2007*

B. Simulation (Virtual Patient)

In Year 4 of the project, our team has delivered two new versions of the Maryland Virtual Patient Environment.

- Including coverage of “unexpected” interventions;
- Allowing discontinued treatments;
- Allowing new diseases to develop due to side effects of treatments.
- The user interface has been redesigned.

- A new agent-based architecture has been developed to support enhanced cognitive capabilities of the virtual patient and the intelligent tutor, including language capabilities.

- In the area of language processing, a dialog processing model was developed.

- Work has continued on improving the language understanding capabilities, centrally including treatment of referring expressions.

- Enhancement of static knowledge resources, the ontology and the lexicon, has been ongoing.

- Work on extending the coverage of diseases has been ongoing: a further improvement of the model of GERD is under way, as is the modeling of cardiovascular diseases.

- A totally reworked system version, with dialog support, is planned for release in June 2008.

- Work has also been ongoing on improving and extending the set of development tools:

  - The DEKADE demonstration, evaluation and knowledge acquisition environment supporting natural language work has been revamped;

  - The interface for creating instances of virtual patients has also been enhanced;

  - A web-based environment for supporting internal documentation has been installed.

C. Smart Image

C.1. Smart Image: CT guided imaging

Objective 1: Dose reduction strategy: Registration between High dose-Low dose CT

- Demonstrated the feasibility of deformable image registration with low-dose CT
- Demonstrated the potential for up to 20-fold reduction in radiation dose

Objective 2: Dose reduction strategy: Iterative reconstruction

- Developed iterative reconstruction algorithm for reconstruction of low-dose CT images. This implementation offers improved image quality at low-dose when compared with scanner-based reconstruction.

- Implemented a parallel version of this algorithm that offers about 25-fold speedup and provides same image quality.

Objective 3: High-speed implementation of non-rigid registration

- Designed and developed an FPGA-based architecture for accelerated implementation of deformable registration algorithm. This architecture is capable of providing 40-fold speedup for image registration.
- Implementation and validation of intensity-based rigid registration using the aforementioned architecture.
- Capability of performing rigid registration (first step to deformable image registration) under 1 minute.

Objective 4: Tracking and visualization

- Developed the mechanism to track the laparoscope and other tools using optical tracking.
- Developed the core components of a visualization system to provide live-augmented reality.

.

C.2. Smart Image: Image Pipeline

● Developed a new method of intra-operative registration that relies on feature-based texture mapping to spatially integrate video sequences into panoramas.

● Performed and published initial technical evaluations on the new registration technique.

● Developed a method of producing “baseline” registration samples using more traditional (camera tracking) procedures against which to compare our new technique.

● Developed a new method of intra-operative depth acquisition dubbed “light fall-off stereo.”

● Initiated collaboration with Stryker Endoscopy to integrate light fall-off stereo into a prototype endoscope.

● Developed design concepts for visualization techniques based on principles of cognitive ergonomics.

● Began evaluation of the design concepts using semi-structured interviews with surgeons.

● Began development of stimulus sequences for use in performance-based user testing.

Reportable Outcomes

A. Informatics

Informatics subgroup 3. Context Aware Surgical Training (CAST)

We have now deployed the prototype system at the MASTRI. Center. Intial Alpha tests were done in one on one sessions with several MASTRI staff and clinicians such as Ivan and Dr. Turner. Additional personnel such as Dr. Godinez, Dr. Sutton, and Sherree were then involved and asked to work with the prototype. Testing is now moving to a beta stage, with a group of volunteer residents that have been assigned by Dr. Godinez. We are noting the feedback. Minor changes or suggestions and bug fixes are being done progressively. Potential new suggestions and major design changes are being logged for discussion with the MASTRI clinicians and staff. In addition, Dr. Jake Seagull is working on a designing a series of experiments to evaluate the performance of our system in terms of

- Improvements in learning outcomes due to Self-feedback
- Improvements in learning outcomes due to Instructor feedback
- Synchronous vs asynchronous feedback

Publications under review

P. Ordóñez, P. Kodeswaran, V. Korolev, W. Li, O. Walavalkar, B. Elgamil, A. Joshi, T. Finin, Y. Yesha, I.George. “A Ubiquitous Context-Aware Environment for Surgical Training”. In Proc. The First International Workshop on Mobile and Ubiquitous Context Aware Systems and Applications (MUBICA 2007), August 2007.

Publications in preparation:

An updated version of that article is currently under preparation for the Journal on Surgical Innovation.

Conclusion

This report began with the recognition that an extraordinary evolution in surgical care has occurred caused by rapid advances in technology and creative approaches to medicine. The increased speed and power of computer applications, the rise of visualization technologies related to imaging and image guidance, improvement in simulation-based technologies (tissue properties, tool-tissue interaction, graphics, haptics, etc) have interacted to advance the practice of surgery. However, the medical profession lags behind other applications of information systems. The research program reported here has proceeded under the mantle of "Operating Room of the Future". As a natural occurrence in the outcome of lessons learned in medicine, we are replacing that theme with the more appropriate "Innovations in the Surgical Environment."

There are three major portions of this study; OR informatics, simulation research, and smart image. Future research efforts will incorporate work related to cognitive ergonomics and human factors as these impact the surgical environment.
\

The purpose of the OR informatics program is to develop, test, and deploy technologies to collect real-time data about key tasks and process elements in clinical operating rooms. We have established testbeds of activities in both simulated and operational environments. We are currently performing tests of the hardware, refining software, and applying lessons learned to hospital operational functions. The objective of Simulation research is to create a system where a user can interact with a virtual human model in cognitive simulation and have the virtual human respond appropriately to user queries and interventions in clinical situations, with a focus on cognitive decision making and judgment. We have made significant strides toward realizing these goals. The MVP simulation functions well for esophageal disorders, and is continuing to expand the repertoire of diseases that are in the simulation model.

The objective of smart image is use real-time 3D ultrasonography and 40-slice highframe-rate computed tomography (CT) for intraoperative imaging to volume rendered anatomy from the perspective of the endoscope. We are combining CT and Ultrasound to overlay image and data to enhance the performance of surgeons-in-training. We have carried out animate model testing of the image registration with great success. We continue to refine and expand our capability through hardware and software refinement.

The year ahead is full of promise for refinements in the use of informatics to support safe and efficient operating room procedures, the use of simulation to improve and accelerate the training of competent surgeons, and the blending of imaging capabilities to provide clearer and safer interactions between patient and surgeon.

Appendices

A. Manuscript: Perioperative Scheduling Group

B. MVP: The Maryland Virtual Patient

C. A Ubiquitous Context-Aware Environment for Surgical Training

D. FPGA-based real-time 3D image preprocessing for image-guided medical Interventions

E. FPGA-Accelerated Deformable Image Registration for Improved Target-Delineation During CT-Guided Interventions

F. A statistical approach to high-quality CT reconstruction at low radiation doses for real-time guidance and navigation

G.. Development of Continuous CT-Guided Minimally Invasive Surgery

H. Development of continuous CT-guided minimally invasive surgery

Appendices

Appendix A:

Manuscript: Perioperative Scheduling Group
Paul Nagy, PhD.

Paul Nagy, PhD.

Abstract

Routine clinical information systems now have the ability to gather large amounts of data that surgical managers can access to create a seamless and proactive approach to streamlining operations and minimizing delays. The challenge lies in aggregating and displaying these data in an easily accessible format that provides useful, timely information on current operations. We describe a Web-based, graphical dashboard that can be used to interpret clinical operational data, allow managers to see trends in data, and help identify inefficiencies that were not apparent with more traditional, paper-based approaches. The dashboard provides a visual decision support tool that also assists managers in pinpointing problem areas in which the greatest benefits can be achieved by using business intelligence techniques to target time and energy toward continuous quality improvement. We review the limitations of paper-based techniques, the development of our automated display system, and key performance indicators in analyzing aggregate delays, time, specialties, and teamwork. Strengths, weaknesses, opportunities, and threats associated with implementing such a program in the perioperative environment are summarized. This research suggests Web-based tools can be made for targeted audiences and adjusted by role, position, or location with results in total participation in quality improvement and constant feedback that provide long-term rewards in cost efficiencies, staff and physician satisfaction, and improved patient outcomes.

Introduction

Management of the modern perioperative environment is a challenging act of balance and orchestration that often tilts perilously close to chaos. Many unforeseen delays (including but by no means limited to patient transport, case cart preparation, consent forms, and slow turnover) can trigger a cascade of events that escalate throughout the day, resulting in frustration for physicians, staff, and patients. The cumulative effects of many small and interacting delays keep the operating room (OR) from running at peak efficiency and can, in some cases, contribute to more serious errors in management and care.

Managers trying to address these delays in an ad hoc fashion find themselves playing “whack-a-mole” with serial problems: as soon as one glitch is resolved, another rears its head. The result is a reactive approach that focuses on immediate problems at the expense of the time and effort needed to identify root causes and long-term solutions that will prevent recurrences. The good news is that various routine clinical information systems now gather enormous volumes of data that surgical managers can access to create a seamless and proactive approach to streamlining operations and minimizing delays.

The process of leveraging these data in support of routine improvements, however, presents its own challenges, particularly in rethinking traditional reporting and analysis techniques. In the past, paper spreadsheet–based reporting methodologies have been used in management meetings, an approach that is insufficient to handle or analyze even the broadest trends in the increasingly large volumes of useful data collected. This traditional, tactical approach most often focuses on only the short-term history of operations and fails to identify the small, recurrent delays that may occur across services.

Transparency and a broad scope of accountability are widely recognized as hallmarks of high reliability and dedication to quality in health care organizations.[1] These values, along with an emphasis on verifiable metrics and automated means of collection and assessment, have figured in significant advances in operations research in the management of the surgical environment in the past 5 years.[2-6] These advances contribute to the fulfillment of important goals of surgical units, including patient safety, access to ORs, economic efficiency, waiting time, and staff satisfaction.[6] Moreover, they have provided novel information about what factors contribute to which specific quality goals. Simulation studies using operations research, for example, have indicated that although immediate quality improvements in patient safety, waiting time, and satisfaction on the day of surgery should be a primary focus, only longer term decisions on staffing will provide economic efficiencies.[6,7] Thus reduction in turnover time in general will not result in increased volume,[8] but access to historical data and application of operations research methods can point to staffing solutions that will optimize economic efficiency.[9]

Our goal, as part of a grant on the OR of the Future from the Telemedicine and Advanced Technology Research Center, was to accelerate the adoption of these advances by providing an automated, holistic view of operations that would enable managers to discover patterns and causes of delays. We created a Web-based, graphical dashboard that could be used to interpret clinical operational data, allow managers to see trends in data, and help identify inefficiencies that were not apparent with more traditional approaches. This dashboard was designed to provide a visual decision support tool that would also assist managers in pinpointing problem areas in which the greatest benefits could be achieved by applying time and energy toward continuous quality improvement.

What is Business Intelligence?

The field of business intelligence, sometimes referred to as business analytics, is the utilization of data warehousing, data mining, modeling, and forecasting to aid in managerial decision support systems.[10,11] Business intelligence is defined as "extracting useful information from the data generated by operational systems of an enterprise."[12] Many top corporate executives use business intelligence–generated electronic scorecarding and dashboarding methodologies to manage their operations with real-time decision making support. A 2007 Gartner, Inc. worldwide survey of 1,400 chief information officers ranked business intelligence as the number one technology priority for remaining strategically competitive.[13] Business intelligence methodologies extend directly to consumers in certain markets. Financial Web sites provide individual investors with extensive research and graphical performance analysis of publicly traded companies.

Large academic medical centers, which often generate revenues in excess of $200 million, are in the same financial league as the medium-to-large businesses in which dashboards are commonplace. Yet few medical centers have invested in the development and routine implementation of tools to analyze and improve the efficiency and effectiveness of perioperative management and operations. Many ORs continue to "fly blind" with regard to concepts such as indexing and performance measurement.

Internal graphical dashboards have been proven in other environments to provide useful and productive platforms for continuous quality improvement. The Six Sigma quality methodology, for example, frequently employs dashboards for process management.[14] A dashboard can provide a consistent framework of defined metrics, known as key performance indicators, that aid in defining and redefining quality and goals as well as offering quantifiable data on achievements.[15] Evidence of consistent improvements through a public dashboard is then used to help align the various parts of an organization to target enhanced performance.

The Potential of Information Visualization

Our project was designed to provide a visual knowledge exploration system to assist managers and senior leadership in understanding trends and patterns. The tools employed in this system provide interactive views of data at various granularities and in a series of graphical or tabular formats. These tools can quickly and with minimal user effort impose various types of analyses on the full dataset or on interactively selected subsets of data.

The goal of a visual knowledge exploration system is to provide tools that facilitate interaction with information in an easy, transparent, and meaningful manner. A well designed graph can tap into the pattern-recognition capabilities of the human visual system. In certain types of patterns, human vision can identify a unique (outlier) value within 200 msec, regardless of whether few or many data points are present.[16,17] However, this ability is entirely dependent on the manner in which the pattern is displayed. Proper visual display is crucial to the use of large datasets for complex decision making support. The optimal type of data display has been the focus of a substantial body of literature and reporting, and the definitive answer changes as rapidly as new technologies enter the information arena.[18-22] Some studies suggest the superiority of graphical formats (bar charts, pie charts, etc.) over tabular presentation (data tables) for certain tasks, whereas the reverse is true for other tasks. More recent work indicates that a constellation of factors must be considered in determining the most advantageous dataset display formats, including type of task, underlying structure of the data, and the knowledge level of the users.[23]

What's the Problem with Paper-Based Reporting?

The benefits of graphical dashboarding can be appreciated more fully by looking at the limitations of traditional paper-based reporting in identifying and managing ongoing operations challenges. Understanding these limitations is important, because in many institutions paper-based reporting is so engrained into routine practice that clinicians and perioperative managers may find it difficult to take the steps needed to adapt to other methodologies. Paper-based reporting management systems are limited in the following areas:

(1) *Time.* Significant time is required to gather information from various sources and compile reports by hand. Decision making is a time-sensitive activity that requires actionable information. Decision making, a process that that should be based on "fresh" data, is adversely affected when time simply does not permit preparation of all possible permutations of analyses that might be informative and useful.

(2) *Effort.* The inherent limitations of paper restrict the number of questions than can be asked and tend to generalize rather than drill down in areas of analyses. Expanding the scope of a paper report requires extra labor. Most often, the result is a trade-off between the time required to generate the report and the quality of effort required in preparing the results for analysis.

(3) *Hindsight.* One of the most frustrating characteristics of paper-based reports is that they provide only answers to questions that were identified *before* the management meeting and discussion. New questions asked during the meeting must be tabled until the next meeting so that analysts can gather the new information required. These tabled questions prevent more purposeful discussions about the data and leave managers with limited information to support decision making in the short term.

(4) *Scope of report.* The amount of information that can be contained in a paper-based report is limited, as is the amount of information that can be reviewed within a reasonable amount of time. Selectivity becomes a necessity; yet it is difficult to predict which questions managers will have during any given meeting. Attempts to broaden the scope of paper-based reports can be both time consuming and problematic: the larger the amount of data in a paper report, the more difficult it is to find any specific piece of information.

(5) *Granularity.* Aggregate statistics do not allow the user to drill down to understand the underlying distributions to evaluate credibility. Mean statistics offered in most paper-based reports are unreliable when describing non-normal distributions of data. A single chart or table on paper can show only one view, and it is difficult to present both overviews and detailed information in a single presentation. Showing trends can obscure source data, where showing only source data can obscure trends. Including both or all in a paper-based report can be labor/time intensive to review and is impractical as a routine practice.

(6) *Multiple versions of truth.* Different groups generating separate reports or even the same reports at different times can result in conflicting operation directives that can

add to confusion in effective decision making. The human intervention inherent in paper-based reporting can also introduce bias that may lead to data analysis errors. Moreover, the passage of time between the collection/analysis of data and final reporting may mean that no direct links exist between current operational data and the paper-based report.

Dashboard Design

In developing the toolkit and dashboard described in this article, we followed the set of principles for effective information seeking outlined by Scheiderman.[24] Effective systems should provide an initial graphical overview of the data, allow the users to zoom in on and filter data in the overview, and then receive details about specific data points on demand. In order to create a our concise visualization environment, we created graphs that were "clickable" within a drill-down interface to provide fast and intuitive zooming and filtering of data. These graphs also provide detailed information about data points when the user hovers over a data marker with the mouse. One of the core requirements for a management dashboard is that it be Web based to allow secure access to all authorized users from any location at any time.

Standard data warehousing techniques were used extraction, transformation, and loading. The database was populated by parsing a text file in a comma-separated delimiter format provided by Cerner SurgiNet Surgical Information System (Kansas City, MO). The text file was converted to ANSI Structured Query Language commands as inserts using the MySQL database administration utility PhpMyAdmin. Data were anonymized for patient information, because the focus of the system was operational efficiency, not identifying specific patient-associated incidents. Six months of operational data were uploaded into the system, incorporating performance statistics on 7,807 cases on 8 MB of disk space on the server. These cases incorporated all of the operating rooms with both inpatient and outpatient admissions.

Identifying Key Performance Indicators

The American Association of Clinical Directors derived a common glossary of the exact meaning of times used for scheduling and monitoring surgical procedures.[25] Time stamps were extracted from the clinical database for: scheduled start time, time at which the patient enters the operating room (PIR), the time at which surgery begins (PST), surgery end time (PF), time at which the patient leaves the room (POR), and the turnover time of the room (TOT). As a hospital policy, when a case begins 15 or more minutes later than scheduled, the circulation nurse must specify a reason for the delay. These performance data are combined with data from the case such as the room, surgeon, anesthesiologist, case number of the day, and the service.

Organization of the Web Site

The Web site was designed to allow analysis from several perspectives. One of the principle mantras of information visualization and data discovery, identified by Schneiderman, is the ability to “overview first, zoom, filter, then details-on-demand.” [26] The ability to view data from multiple perspectives assists and increases confidence in decision making. The user accesses each category via a navigation bar of tabs at the top of the web site. As the user navigates through the system, a trail (called a “breadcrumb” and shown in Figure 1) is displayed to illustrate how the user navigated to that point and to allow easy backtracking.

Aggregate Delay Analysis

A total of 43 delay types were identified as reasons for delays. These were grouped into general root causes of materials, patient, prerequisite task(s), scheduling, staff, and transport. At the top of the delay analysis page, as shown in Figure 1, pie charts demonstrate the relative number of delays per root cause and their cumulative impact in time. Although some delays are not numerous, they might have a large effect on the operations of an OR. The user clicks on the pie chart, selects a root cause, and is presented with an analysis page of all the underlying delay causes for that root cause, broken down in the same way by relative number and impact. By selecting a delay cause, the system moves to show a breakdown by specialty, displaying the number of incidents and their average delay times. Selecting a service displays all the cases, and by selecting a case the details for that case can be displayed. Within the span of 4 clicks, a user can drill down from all the cases in the database to the details of an individual case. The delay analysis tool is useful in understanding the cumulative cost of systemic delays and which specialties are most affected by them.

Temporal Analysis

The temporal perspective provides a daily tactical review of cases to determine over a specified period of time which ones were delayed and why. To present the utilization levels of the ORs in a given day, we used a polar chart showing cumulative room utilization as a function of the hour of the day (Fig. 2). This is useful for look at the relationship between room utilization and staffing levels. The user can drill down to the specifics of a single case or choose to look at data grouped by room or specialty. System delay types, such as transport issues, can affect multiple rooms and specialties across suites of ORs over different periods of time.

Service Analysis

Service analysis focuses on key performance indicators within each specialty. Medical specialties within the OR have widely differing dynamics for case efficiency, utilization, turnover, and case length based on a number of factors, including but not limited to procedure complexity and patient acuity. For some types of data analysis involving services or subspecialties, bubble charts provided a useful way to organize data (Fig. 3). The bubble chart plots each service by its average case length in the *x* axis and average delay duration in the *y* axis. The size of the bubble for each specialty is directly proportional to the number of cases performed. The more cases a service performs, the larger the diameter of the bubble.

For each specialty analysis, the site provides histograms for case length and delay duration. Histographic analysis is useful in determining the distribution type, the spread of the distribution, and the existence of outliers that may distort statistical analysis.

Another display generated was a scatter graph of all cases plotted by their scheduled case lengths compared with actual duration (Fig. 4). Regression analysis shows potential correlations, along with graphical bands illustrating confidence intervals for the line and the points. This index of predictability of scheduling is especially useful in identifying and drilling down on the outlier cases to understand their causes of variance. Each diamond represents an individual case, and, by clicking on a diamond, the details of that case are displayed (Fig. 5).

Teamwork Analysis

With a surgeon and principal anesthesiologist assigned to each case, we can display delay causes for those cases. As shown in the spider graph in Figure 6, delays are grouped by and aggregated by the blue bars. The farther out the bars, the greater were the number of occurrences for that root cause. In an overlapping orange are the average delays by root cause for all physicians in that specialty, normalized by the number of procedures done by that physician.

Using this teamwork analysis, it is possible to identify specific teams that appear to work well together as well as those that are not routinely time efficient. Other factors, of course, must be considered in reviewing these data, and it would be difficult to determine root causes for efficiency or inefficiency in a specific case. However, this knowledge may provide strategic information that could contribute to what business intelligence experts call a “discovery cascade.”

Results

Results of an initial rollout of the Web system were assessed through interviews with senior management. This included discussions with the chief medical officer, chief operations officer, chief nursing officer, chairs of surgery and anesthesiology, and several perioperative managers. In presenting this potentially disruptive tool to management, we performed a strengths, weaknesses, opportunities, and threats analysis (known in business intelligence parlance as a SWOT analysis) to classify their observations.

Strengths

Our Web-based approach was seen as a powerful tool that would aid management in identifying systemic, process-driven root causes for delays and other problems and that had the potential for positive effects on the culture of the organization. Among the positive aspects they cited were: (1) This approach turns traditional paper-based data into knowledge and presents this knowledge in easy to digest chunks. (2) The dashboard is independent of any single vendor. (3) If used with a data repository, it has the potential ability to link data from different information systems. (4) It provides a systemic view that can calculate the total costs of root causes. (5) It provides a quick visual way to target improvement. (6) Additional metrics can be added at the request of management with minimum programming effort. (7) Visual displays made outliers and trends more easily identifiable and rendered distributions more easily understood than standard aggregate statistics.

Weaknesses

The reviewers identified 4 areas of weakness and potential improvement for the Web-based system. (1) Timeliness: Depending on the method of data extraction, data may not be live or near-live. If data are provided via an upload, they will be only as recent as the last event. The dashboard optimally should have an Open Database Connectivity connection to a clinical data repository (CDR) or similar copy of the live production environment. The update schedule of the CDR will determine the timeliness of the dashboard.

(2) Personnel resources: Skilled personnel are required to build and maintain a graphical dashboarding application. A surgical informaticist is a good choice as they have the clinical domain experience combined with the principles of management and information technology. This person needs to guide the development of metrics by using clinical knowledge to extract meaningful and relevant data. This individual can also play a crucial role in bridging the cultures among health care providers, information technology specialists, and business process managers.[27]

(3) Hardware resources: Hardware resources include access to server space with sufficient processing power and storage to handle a large database. The database storage space required is minimal; however, the central processing unit that drives the data mining must be powerful.

(4) Management training: The introduction of analytics and acceptance of business intelligence practices within a group, particularly one that already has a long-engrained operations process, cannot be accomplished overnight.[11] An investment of time and effort is required and involves education of managers on the use of these tools and the ways in which they can be incorporated into decision making. In the process, the focus of the managers and the entire organization should change from trying to understand the latest event to looking at trends within the data to predict what will happen next and identify ways to achieve the best possible results.

Opportunities

A dynamic surgical block utilization chart with easily available drilldown, as created in our project, allows surgical chiefs to continually monitor utilization. The drilldown permits them to see which days are being underutilized and by whom and points to immediate courses of action rather than waiting for end-of-year retrospective and analysis. When competition is fierce for OR time, this transparency can be extraordinarily valuable for surgical practices.

Another potential benefit that can accrue from matching staffing with caseload to optimize OR efficiency.[28] Cases in overutilized time are 1.75 times more expensive than cases during normal staffing hours[29], the goal is to match case load with staff.[29] The dashboard tool pulls in scheduling data from the clinical information system and can display the number of projected cases at 1-hour intervals. The dashboard can also use retrospective data on add-on cases to estimate a caseload probability by the hour. To maximize efficiency, a user input can be created whereby a manager or charge nurse may enter data on staffing levels, helping to match case load to staffing.

Along with scheduling efficiency, OR senior staff and managers may want to match clinical proficiencies with cases. Displays can be created to show circulator/scrub combinations with surgical specialties and case types similar to the surgeon−anesthesia graphs presented. This gives managers opportunities to maximize good teams and identify teams that need improvement.

Many opportunities are available for benchmarking between services and between organizations. The only limitation is the ability to capture data from an information system or network, apply a meaningful analysis, and provide an easily understood graphic for the appropriate audience.

Current Procedure Terminology (CPT) codes would be an important additional piece of information, because case efficiency should be benchmarked against similar cases. Cardiac Thoraic cases, for example, have long turnover times because of the degree of complexity involved in setup of equipment, drawing of drugs, patient preparation, etc. National benchmarking can be imported to compare against the organization’s benchmarks reports for CPT and DRGs, morbidity and mortality, length of stay, and complications and presented in an easy-to-navigate and -understand visual.

Opportunities for assessing clinical outcomes include measures such as infection control, preoperative antibiotic compliance, unplanned returns to the OR, staff compliance on chart quality, timeliness, completeness, staff arrival time, etc. Financial reports can include direct costs, indirect costs, contribution margins per case/specialty, labor costs, supply costs per case, and metrics associated with defining and monitoring best practices.

Threats

Two potential threats to a system such as the one we devised were identified by the interview group and by our own developers.

The first threat is in the area of data quality and integrity. The data retrieved for our clinical information system have 2 sources of origin. First, scheduling data are obtained. These include but are not limited to scheduled start date and time, duration, procedure, surgeon, and anesthesiologist. The schedulers at our institution reside both centrally in a surgical posting office and decentralized in physicians' offices (in the oral maxillofacial and organ transplant services). Because scheduling data do not directly enter the patient's medical record or roll directly into clinical documentation that must be reviewed and modified by a nurse, it is assumed that the risk for bias is minimal. Manipulation of case durations and scheduled start times is limited by system controls. The surgical posting office does have the ability to override system data (e.g., for scheduled case duration, which is a byproduct of historical averages), but this is not done without approval from a supervisor.

Data from nursing documentation is under constant review by various clinicians to audit work. This process ensures data integrity and compliance and serves as a modest check and balance. Most of these data are objective, and although some bias may be present, this will most likely be minimal. The area of documentation that is most prone to bias is the "delay reason," because of its highly subjective nature and possible repercussions from management. Another factor for inaccurate delay reporting is the phenomenon of cascading delays (i.e., when a delay early in the day causes delays in subsequent cases). By the time of the third or fourth delayed case, it is difficult to ascertain the cause other than to note that the previous case "ran over." During this series of delays, an entirely different cause of delay may happen in a specific case, but the reference point of a scheduled start time is lost, so that it is much more difficult to document a delay cause and duration. Time stamps (in-room time minus scheduled start time), of course, provide well documented and precise record of delay in minutes, but this does not qualify delay by reason type and provides no insights for root cause analysis.

The second threat to initiation of a system such as the one we developed lies in the general perceptions by staff and physicians. Many may feel that they are being spied upon or monitored, especially in areas in which no previous metrics existed. Others may find themselves out of their routine comfort zones. Underperforming staff who worry that they may be identified by the system may aggressively resist implementation of the new tools or work to undermine data integrity. Depending on the organization's structure, the open availability of data could result in "punishment" for individuals or a group rather than the intended promotion of positive departmental and institutional change. Moreover, in environments in which competition for OR time is strong, surgical chiefs may be tempted to use data as a weapon to promote their own agendas.

The transparency of the data should alleviate some of these concerns. Team members should be able to see the data in which performance is being judged. In the past, data was obtained by someone walking around with a clipboard or in a back office recording data off charts, with limited or no ways to verify whether or the data were true and accurate. With the drilldown features and different ways of organizing data for dashboard display, team members can easily view the raw data.

Conclusion

Strategic decisions made on the basis of management instinct have a lasting impact on the well-being of an organization. Management could benefit from the adoption of business intelligence tools that provide a quantifiable, validated alternative to instinct and ad hoc choices decision making.

Behavior and practice changes are central to achieving the objective of quality reports that drive efficiency. Too often data are not integrated within the scope of daily practice. Acceptance of the importance of data must become a part of the culture of the organization. Graphical dashboards that present information down to the simplest, easiest-to-understand, and most accurate levels can compel this behavior change. Managers in the perioperative environment should seize the opportunity to integrate data into their organizations' cultures. Our research suggests that one promising approach is in Web-based tools that can be made for targeted audiences and adjusted by role, position, or location. The result can be total participation in quality improvement and constant feedback that provides long-term rewards in cost efficiencies, staff and physician satisfaction, and improved patient outcomes.

Acknowledgements

This project was supported under an OR of the Future grant from the Telemedicine and Advanced Technology Research Center. I would like to thank Dr. Nancy Knight from the University of Maryland for her expert assistance in preparing this manuscript.

References

1. Weick KE, Sutcliffe KM: Managing the Unexpected: Assuring High Performance in an Age of Complexity. San Francisco, CA: Jossey-Bass, 2001.
2. Dexter F, Xiao Y, Dow AJ, Strader MM, Ho D, Wachtel RE: Coordination of appointments for anesthesia care outside of operating rooms using an enterprise-wide scheduling system. Anesth Analg 105:1701-1710, 2007.
3. Dexter F: Why calculating PACU staffing is so hard and why/how operations research specialists can help. J Perianesth Nurs 22:357-359, 2007.
4. Dexter F: Bed management displays to optimize patient flow from the OR to the PACU. J Perianesth Nurs 22:218-219, J2007.
5. Dexter F: Operating room utilization: information management systems. Curr Opin Anaesthesiol 16:619-622, 2003.
6. Dexter F, Epstein RH, Traub RD, Xiao Y: Making management decisions on the day of surgery based on operating room efficiency and patient waiting times. Anesthesiology 101:1444-1453, 2004.
7. Macario A, Chow JL, Dexter F: A Markov computer simulation model of the economics of neuromuscular blockade in patients with acute respiratory distress syndrome. BMC Med Inform Decis Mak 6:15, 2006.
8. O'Sullivan CT, Dexter F, Lubarsky DA, Vigoda MM: Evidence-based management assessment of return on investment from anesthesia information management systems. AANA J 75: 43-48, 2007.

9. O'Neill L, Dexter F: Tactical increases in operating room block time based on financial data and market growth estimates from data envelopment analysis. Anesth Analg 104: 355-368, 2007.
10. Davenport TH: Competing on analytics. Harvard Bus Rev. January 2006:1-12.
11. Davenport TH, Harris JG: Competing on analytics: the new science of winning. Boston, MA: Harvard Business School Press, 2007.
12. Chisholm M: The twin towers of BI babel: enterprise architecture. BI Rev. December 2007. Available at: www.bireview.com/issues/2007_42/10000440-1.html. Accessed: January 10, 2008.
13. Beer S: Business intelligence top priority of CIOs. itWire. February 2007. Available at: www.itwire.com.au/content/view/9906/53/. Accessed: January 10, 2008.
14. Pande PS, Neuman RP, Cavanagh RR: The Six Sigma Way: Team Fieldbook. New York, NY: McGraw-Hill, 2002.
15. Malik S: Enterprise Dashboards: Design and Best Practices for IT. Hoboken, NJ: John Wiley & Sons, 2005.
16. Treisman A: Preattentive processing in vision. Comput Vis Graphics Image Processing 31:156-177, 1985.
17. Treisman A, Gormican S: Feature analysis in early vision: Evidence from search asymmetries. Psychol Rev 95:15-48, 1988.
18. Washburne JN: An experimental study of various graphic, tabular, and textual methods of presenting quantitative material. J Educ Psychol 18:361-376, 465-476, 1927.

19. Tufte ER: The Visual Display of Quantitative Information. Cheshire, CT: Graphics Press, 1983.

20. Cleveland WS, McGill R: Graphical perception and graphical methods for analyzing scientific data. Science 229:828-833, 1985.

21. Montazemi AR, Wang S: The effects of modes in information presentation on decision-making: a review and meta-analysis. J Manag Inform Syst 5:101-127, 1988.

22. Feldman-Stewart D, Brundae MD, Zotov V: Further insight into the perception of quantitative information: Judgments of gist in treatment decisions. Med Decis Making 27:34-43, 2007.

23. Meyer J, Shamo MK, Gopher D: Information structure and the relative efficacy of tables and graphs. Hum Factors 41:570-587, 1999.

24. Card SK, Mackinlay JD, Shneiderman B: Readings in information visualization: using vision to think. San Francisco, CA: Morgan Kaufmann Inc., 1999.

25. Procedural Times Glossary of the AACD. AACD. October 2005. Available at: aacdhq.org/Glossary.htm. Accessed: January 10, 2008.

26. Shneiderman B: Inventing discovery tools: Combining information visualization with data mining. Informat Visualiz 1:5-12, 2002.

27. Charters KG: Nursing informatics, outcomes, and quality improvement. AACN Clin Issues 14:282-294, 2003.

28. Dexter F, Ledolter J, Wachtel RE: Tactical decision making for selective expansion of operating room resources incorporating financial criteria and

uncertainty in sub-specialties' future workloads. Anesth Analg 100:1425-1432, 2005.

29. Strum DP, Vargas LG, May J: Surgical subspecialty block utilization and capacity planning: A minimal cost analysis model. Anesthesiology 90:1176-1185, 1999.

Appendix B:

MVP: The Maryland Virtual Patient

Project Report

Dr. Bruce Jarrell
Dr. Sergei Nirenburg
Dr. Marjorie McShane
Dr. Stephen Beale
Dr. George Fantry

February 28, 2008

Summary

In Year 4 of the project, our team has delivered two new versions of the Maryland Virtual Patient Environment. The realism of the simulation has been enhanced by including coverage of "unexpected" interventions; allowing discontinued treatments; allowing new diseases to develop due to side effects of treatments. The user interface has been redesigned. A new agent-based architecture has been developed to support enhanced cognitive capabilities of the virtual patient and the intelligent tutor, including language capabilities. In the area of language processing, a dialog processing model was developed. Work has continued on improving the language understanding capabilities, centrally including treatment of referring expressions. Enhancement of static knowledge resources, the ontology and the lexicon, has been ongoing. Work on extending the coverage of diseases has been ongoing: a further improvement of the model of GERD is under way, as is the modeling of cardiovascular diseases. A totally reworked system version, with dialog support, is planned for release in June 2008. Work has also been ongoing on improving and extending the set of development tools – the DEKADE demonstration, evaluation and knowledge acquisition environment supporting natural language work has been revamped; the interface for creating instances of virtual patients has also been enhanced; a web-based environment for supporting internal documentation has been installed. Finally, we have written, submitted, published or delivered X conference and journal papers.

This report introduces the basics of the MVP approach, discusses its place on the map of intelligent systems in clinical medicine and describes the project's status and research and development activity presently under way.

Background

Many medical educators believe that the current system of medical education in the US fails to reliably provide students with a sufficient breadth of clinical experience to ensure the development of clinical diagnosis and treatment skills. Students typically manage too few patients with too few clinically relevant variations of a disease to become first-rate care providers by the time they complete their residency. In addition, learning clinical medicine through experience on live patients imposes a heavy responsibility, offering no channel for learning by trial and error. One way to circumvent these shortcomings of medical education is through interactive computer systems that simulate a clinical care environment.

The specific goal of the MVP project is to create a computer system for teaching medical students cognitive skills of an attending physician related to diagnosing and treating patients. The system is intended to provide a safe, hands-on environment in which the students communicate with and treat simulated virtual patients as well as obtain help from an automatic mentor. The system environment also approximates the real world in making the student a member of a team of medical professionals – simulated lab technicians and specialist consultants.

For a system of this kind to be successful, it must be **realistic** in a variety of ways. Diseases and their progression must be realistically simulated, using biomechanistic knowledge whenever possible but reverting to expert clinical knowledge in situations where biological mechanisms are still unknown. Virtual patients must be capable of carrying more than one disease at a time. Side effects of treatments must be modeled to allow dynamic changes in the pathophysiological states of virtual patients. The above implies that virtual patients cannot be fully pre-scripted, which means that simulations must be based on causal modeling of event sequences. Virtual patients must be able to model symptom perception, communicate with the physician and make reasoned decisions relating to treatment. Mentoring knowledge should reflect the best clinical practices in diagnosis and treatment. A broad inventory of diseases must be covered. Finally, even for a single disease, a pedagogically sufficient number of virtual patients must be created (or allowed to be created) with different genetic predispositions, disease progressions and reactions to treatment.

To meet the above requirements, a computer system must have several **enabling capabilities**. The simulation module must be able to deal with knowledge of different provenance and varying grain size, on a scale from, e.g., well-understood cell-level biochemical mechanisms to, e.g., qualitative expert experience with treatment outcomes or co-morbidities. The virtual patient must be capable of perception, reasoning and action. This makes it a classical example of an artificial cognitive agent. Language is the most natural way of communicating with the human user of the system and the only truly realistic one. Finally, the amount and variety of medical knowledge required for building a realistic system is non-negligible.

To maintain realism, the system must be meaningfully responsive in novel, unexpected, unscripted situations. The main source of this novelty is unpredictability of the actions of the human (the medical student). There are two ways in which the human can introduce novel situations – through treatment interventions and in language-based communication with the virtual patient and the tutor.

For a system of this nature to be **feasible**, a number of constraints must be introduced in its design. It is clear that the coverage of medical knowledge must be incremental, starting with a subset of diseases and gradually increasing their inventory. The grain size of disease description can also start with a minimally useful level of coarseness and made progressively finer over time in successive versions of the system. The simulation of auxiliary agents – lab technicians and specialist consultants – can be at first skeletal. It is possible initially to exclude direct visual and haptic physical examination of the patient. The language communication can be in written form, not speech-based. However, the virtual patient and the tutor must be able to understand the content of the communication and the intentions of the human user encoded in written messages and be able to reason about how to respond appropriately. If communication is to be realistic, it will not be possible to have this understanding rely on a predetermined list of expected questions with their answers. Similarly, while it is possible to limit the set of intentions, goals, plans, character traits and action types modeled in the virtual patient, the overall mechanism of manipulating them must be developed. This is because neither pre-scripting the virtual patient's actions in response to a variety of human interventions nor determining what to do stochastically can guarantee sufficient realism in system behavior.

Why do we believe that our work is feasible?

First, our development efforts are targeted toward **specific applications**: there is no attempt to develop a fully generalized, plug-in ready cognitive architecture (like TRAINS/TRIPS), or to implement a broad-coverage, domain-independent dialogue system, or to equip system agents with all of the plans and goals of human beings, or to endow them with the full spectrum of possible character traits (as is done in theoretical approaches to affective modeling), or to model diseases at a grain size any finer than that needed to support the given application. Instead, theoretical and practical advancements are geared toward the near- and long-term future of the specific systems, with infrastructure decisions being made with a long-term view but knowledge support targeted at near-term goals.

Second, the **integrated approach to knowledge modeling** in MVP permits the same ontological substrate to be used for knowledge-based simulation, planning, and NLP, meaning that once knowledge is encoded it is available to all system agents and processors. The OntoSem ontology used in MVP already includes over 9,000 concepts, described by an average of 16 properties each; around 7,500 of those are from the general domain, with the remaining 1,500 devoted to medicine. Moreover, since scripts describing complex physiological and cognitive events are formally part of the ontology, the same scripting language used for physiological simulation (which is already understood by our simulator engine) can be used for planning and dialogue.

Third, the dialogue processing model is grounded in the **OntoSem deep semantic natural language processing system,** which has been under development for over 20 years.

Fourth, the past decade has produced a valuable **body of research** on cognitive engineering, agent networks, planning, plan- and goal-centered dialogue systems, etc. This large body of work includes inventories of needs for intelligent systems, sample architectures, descriptions of problems encountered, bridges between descriptive, theoretical and implementational work, and reports from the field that provide a good understanding of the current state of the art. In short, this body of work is permitting us to quickly reap the benefits of hard-won insights.

The MVP project involves a broad variety of specific tasks. Knowledge about a) human physiology and pathophysiology, b) best clinical practices and c) specific instances of virtual patients must be acquired from experts. This knowledge must be represented in a way that facilitates automatic processing by computer. Methods of computer processing of this data must be developed, including a) disease progression simulation, b) reasoning capabilities to simulate the virtual patient's and virtual tutor's decision making, and c) communication capabilities between people and artificial agents. Finally, issues relating to a) software system architecture and maintenance; b) static knowledge resource maintenance; and c) testing and evaluation support must be addressed.

In what follows, we will describe the current status and immediate plans of the MVP project.

The Knowledge and Processing Architecture

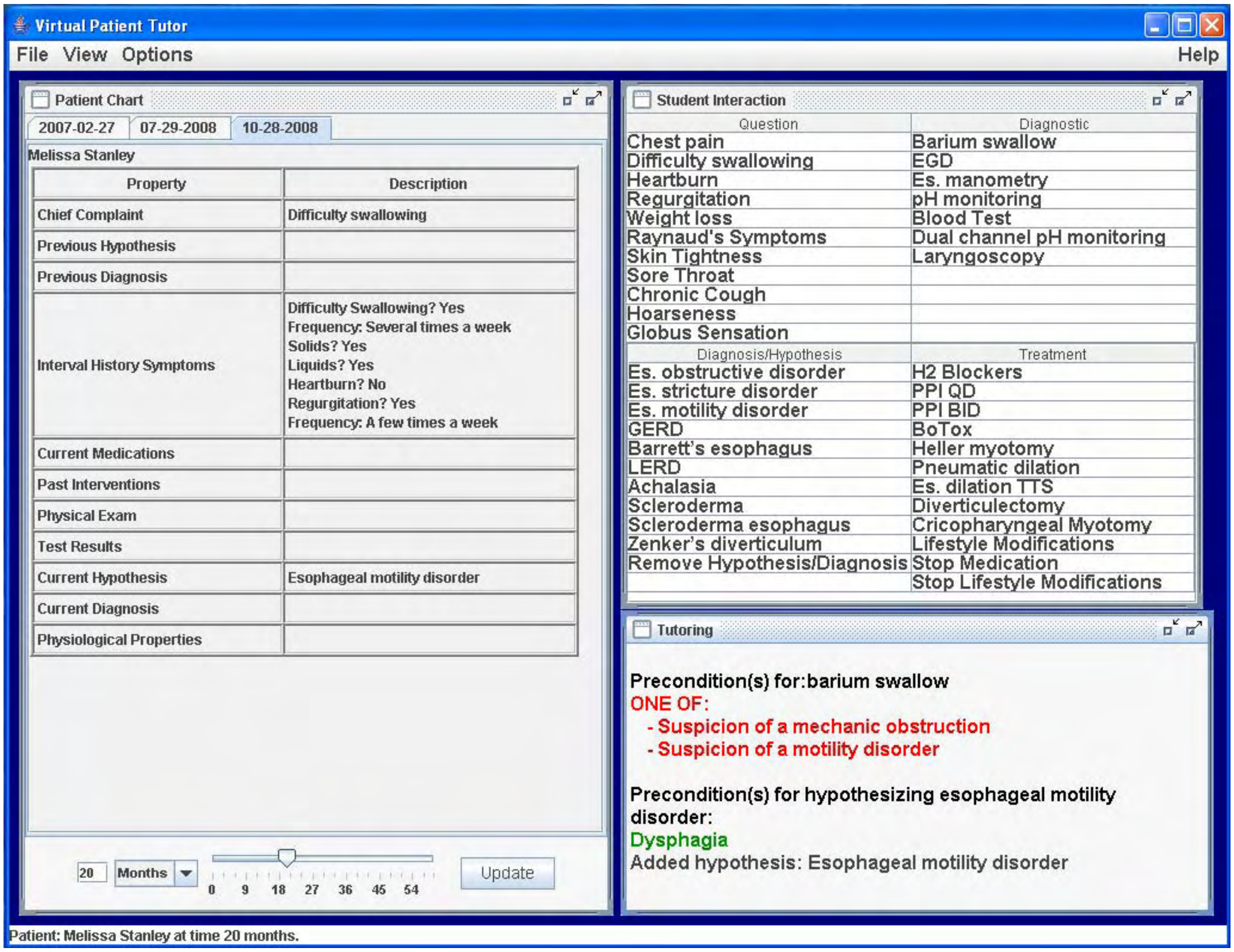
Current Functionality of MVP

We present here a simplified, coarse-grained sketch of the MVP simulation, interaction and tutoring system (further details will be supplied later). A virtual patient instance is launched and starts its simulated life, with one or more diseases progressing. When the virtual patient develops a certain level of symptoms, it presents to the attending physician, the system's user.[5] The user can carry out, in an order of his or her choice, a range of actions: interview the patient, order diagnostic tests, order treatments, and schedule the patient for follow-up visits. The patient can also automatically initiate follow-up visits if its symptoms reach a certain level before a scheduled follow-up. This patient-physician interaction can continue as long as the patient "lives."

[5] Human users of the MVP system can be of various profiles, including medical students, residents, physicians or other medical personnel seeking to refresh or improve their skills in some area, physicians (both developers and non-developers) testing the system for accuracy and robustness, and examinees. Throughout this paper this heterogeneous group of users will typically be referred to simply as "users".

As of the time of writing, the implemented MVP system includes a realization of all of the above functionalities, though a number of means of realization are temporary placeholders for more sophisticated solutions, currently under development.[6] The most obvious of the temporary solutions is the use of menu-based patient-user interaction instead of natural language interaction. While this compromise is somewhat unnatural for our group, which has spent the past 20 years working on knowledge-based NLP, it has proved useful in permitting us to focus attention on the non-trivial core modeling and simulation issues that form the backbone of the MVP system.

MVP currently covers six esophageal diseases pertinent to clinical medicine: achalasia, gastroesophageal reflux disease (GERD), laryngopharyngeal extraesophageal reflux disease (LERD), LERD-GERD (a combination of LERD and GERD), scleroderma esophagus and Zenker's diverticulum.[7]



Melissa Stanley

Property	Description
Chief Complaint	Difficulty swallowing
Previous Hypothesis	
Previous Diagnosis	
Interval History Symptoms	Difficulty Swallowing? Yes Frequency: Several times a week Solids? Yes Liquids? Yes Heartburn? No Regurgitation? Yes Frequency: A few times a week
Current Medications	
Past Interventions	
Physical Exam	
Test Results	
Current Hypothesis	Esophageal motility disorder
Current Diagnosis	
Physiological Properties	

Question	Diagnostic
Chest pain	Barium swallow
Difficulty swallowing	EGD
Heartburn	Es. manometry
Regurgitation	pH monitoring
Weight loss	Blood Test
Raynaud's Symptoms	Dual channel pH monitoring
Skin Tightness	Laryngoscopy
Sore Throat	
Chronic Cough	
Hoarseness	
Globus Sensation	

Diagnosis/Hypothesis	Treatment
Es. obstructive disorder	H2 Blockers
Es. stricture disorder	PPI QD
Es. motility disorder	PPI BID
GERD	BoTox
Barrett's esophagus	Heller myotomy
LERD	Pneumatic dilation
Achalasia	Es. dilation TTS
Scleroderma	Diverticulectomy
Scleroderma esophagus	Cricopharyngeal Myotomy
Zenker's diverticulum	Lifestyle Modifications
Remove Hypothesis/Diagnosis	Stop Medication
	Stop Lifestyle Modifications

Precondition(s) for:barium swallow
ONE OF:
- Suspicion of a mechanic obstruction
- Suspicion of a motility disorder

Precondition(s) for hypothesizing esophageal motility disorder:
Dysphagia
Added hypothesis: Esophageal motility disorder

Patient: Melissa Stanley at time 20 months.

Figure 1. The main user interaction screen.

[6] The motivation for the use of the placeholders was the desire for rapid integration of a working prototype of the system. This goal was successfully attained, and at the time of writing the MVP has started to be used in the curriculum of the University of Maryland School of Medicine.

[7] Detailed descriptions of the diseases achalasia and GERD are given in McShane et al. *forthcoming* and McShane et al. *submitted*.

Figure 1 shows a screen shot of the main pane of the simulation and mentoring application. The left-hand side is the patient chart, with the page for the current visit presented. Records of previous visits can be reached using their associated dated tabs. The chart includes both information that is automatically generated from the previous visits and information that is compiled based upon the events that occurred during the latest visit. The information carried over from previous visits includes the chief complaint, the (previous) disease hypothesis and diagnosis, currently prescribed medications and past interventions. The information that is added based on the latest visit includes current symptoms (called Interval History Symptoms), results of the (simulated) physical exam, results of tests ordered during the given visit, and a disease hypothesis and/or diagnosis added during this visit. Depending on the specific tutoring setting, the user can be given access to a complete list of the virtual patient's physiological properties (the omniscient view of the patient).

The lower left pane contains a time slider, which permits the user to advance patient time at intervals of days, weeks or months. Other configurations of the system include other methods of managing simulation time.

The upper right quadrant contains four panes of menus with which the user can ask the patient relevant questions about symptoms, order diagnostic tests, posit a hypothesis and diagnosis, and order treatments.

- Each **symptom** button asks all relevant questions about the given symptom at once: e.g., asking about heartburn queries the patient about whether he/she ever has heartburn and, if so, follows up with questions about frequency, severity, and whether any kinds of food cause heartburn.[8]
- Each **diagnostic** button orders that diagnostic test, with the results appearing as fillers of Test Results in the patient chart.
- Each **diagnosis/hypothesis** button permits the user to select that disease as a diagnosis or hypothesis, with the associated slots of the patient chart being filled accordingly.[9] For system configurations that include automatic tutoring, the recording of hypotheses and diagnoses – which make explicit the user's current thinking – is needed for the evaluation of whether a move is being made for the right reason.
- Each **treatment** button carries out a treatment, with the treatment then recorded in the Past Interventions cell of the current and subsequent patient charts.

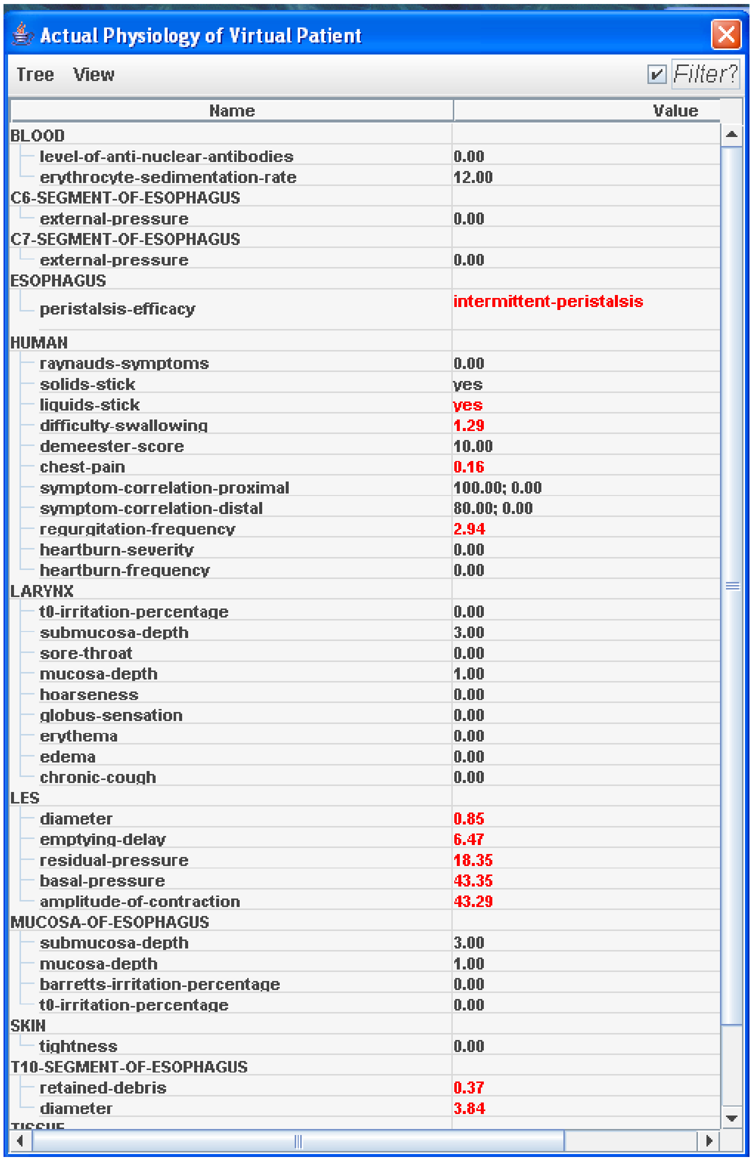
The lower right quadrant contains mentoring messages for configurations when the tutor agent is enabled. (See section 2.7 for further discussion.)

Returning to the patient chart, it can provide access to the omniscient view of the patient's physiological properties, a feature that can be enabled and disabled under various system configurations. For example, Figure 2 shows a filtered subset of the

[8] When physicians and students tested the system, the preference for asking nests of questions with a single click was overwhelmingly confirmed. When natural language support is added, questions will be asked in series, mimicking office interviews.

[9] After the initial choice of a disease from a menu, a pop-up box asks whether this is a diagnosis or a hypothesis.

properties of Melissa Stanley 20 months into her disease. The values in red are those that changed since the last time the clock was advanced. This physiological view of the virtual patient is useful not only for validating system functionality, it also has pedagogical uses: a user can, under certain configurations that the instructor can control, watch property values change in response to various interventions. This physiological view drives home the point that the MVP system is not an inventory of stored scenarios: it is a live simulation whose outcomes are crucially dependent upon user actions.



Name	Value
BLOOD	
level-of-anti-nuclear-antibodies	0.00
erythrocyte-sedimentation-rate	12.00
C6-SEGMENT-OF-ESOPHAGUS	
external-pressure	0.00
C7-SEGMENT-OF-ESOPHAGUS	
external-pressure	0.00
ESOPHAGUS	
peristalsis-efficacy	intermittent-peristalsis
HUMAN	
raynauds-symptoms	0.00
solids-stick	yes
liquids-stick	yes
difficulty-swallowing	1.29
demeester-score	10.00
chest-pain	0.16
symptom-correlation-proximal	100.00; 0.00
symptom-correlation-distal	80.00; 0.00
regurgitation-frequency	2.94
heartburn-severity	0.00
heartburn-frequency	0.00
LARYNX	
t0-irritation-percentage	0.00
submucosa-depth	3.00
sore-throat	0.00
mucosa-depth	1.00
hoarseness	0.00
globus-sensation	0.00
erythema	0.00
edema	0.00
chronic-cough	0.00
LES	
diameter	0.85
emptying-delay	6.47
residual-pressure	18.35
basal-pressure	43.35
amplitude-of-contraction	43.29
MUCOSA-OF-ESOPHAGUS	
submucosa-depth	3.00
mucosa-depth	1.00
barretts-irritation-percentage	0.00
t0-irritation-percentage	0.00
SKIN	
tightness	0.00
T10-SEGMENT-OF-ESOPHAGUS	
retained-debris	0.37
diameter	3.84
TISSUE	

Figure 2. A subset of “Melissa Stanley”’s property values at month 20.

An Example of Patient Management

To illustrate the system operation, we present below an example of how a user might handle the case of “Melissa Stanley,” the patient a subset of whose profile is shown in Figures 1 and 2. The sequence of steps in this example represents only one of thousands of possible paths through this patient's case. Indeed, different users, working under different tutoring intensity settings, will create very different system runs. At the start of this particular run the tutor agent is turned on to the maximal information-providing tutoring setting. With this setting, the tutor agent reveals, at appropriate times, all fulfilled and unfulfilled preconditions (marked in green and red, respectively) for user actions, and user actions are only permitted if the necessary preconditions are fulfilled. In this illustration, the agent of the action at each step is *italicized*, and the action itself is marked in **boldface**.

1. The *VP instance*, Ms. Stanley, **presents** with the chief complaint "difficulty swallowing". The time of presentation – 17 months into the disease – is indicated by the position the time slider but can be hidden if desired for pedagogical reasons.
2. The *user* **asks** about heartburn, difficulty swallowing and regurgitation.
3. The *VP instance* (Ms. Stanley) **responds** stating that the only available positive symptom is mild difficulty swallowing occurring less than once a week (the patient has the personality trait of being a hypochondriac, meaning that she presents to the doctor given very mild symptoms).
4. The *user* decides that the symptoms are not severe enough to be acted upon and **schedules** a follow-up visit in 3 months. This ability to monitor a patient over time, with or without intervention, is a crucial aspect of clinical medicine not targeted by any other simulation systems.
5. The clock advances to 20 months, and the *VP instance* **presents** for the scheduled visit.
6. The *user* **asks** about swallowing again as well as about regurgitation and heartburn.
7. The *VP instance* **responds** that difficulty swallowing and regurgitation both occur several times a week; there is no heartburn.
8. The *user* **orders** a barium swallow diagnostic test
9. The *tutor agent* **blocks** the action, **responding** with the following message (in red) to the user:

   Precondition(s) for barium swallow

   ONE OF:

   - SUSPICION OF A MECHANIC OBSTRUCTION
   - SUSPICION OF A MOTILITY DISORDER

10. The *user* **posits** 'motility disorder' as the hypothesized disorder (using the diagnosis/hypothesis button for 'Esophageal motility disorder').
11. This action is **permitted** by the *tutor* because its preconditions have been satisfied (namely, the patient is known to have dysphagia, which is difficulty swallowing). Figure 1 shows the view of the user's interface at this point in the scenario.
12. The *user* again **orders** a barium swallow diagnostic test.
13. This time the *tutor* **permits** this action.
14. A *lab technician agent* **produces** numerical results for this test, **passes** them on to a *specialist agent*, who **returns** the results and an interpretation: "Subtle narrowing of LES. No dilated esophagus." The result and the interpretation are recorded in the patient chart.
15. The *user* decides that it is still too early to intervene, sends the patient home and

schedules another follow-up visit in 12 months.

16. When the *VP instance* **presents** at 32 months, difficulty swallowing and regurgitation have increased in frequency and severity, as the user can see by comparing the values for this visit to the values from the last visit (accessible by the dated tabs).
17. The *user* **orders** the diagnostic tests EGD, barium swallow and manometry, in that order.
18. The *tutor* **permits** all of these moves and, since maximum tutoring support is enabled, the preconditions for each of these actions are shown when the user carries out the action, reinforcing the teaching points related to prerequisites for good clinical practice.
19. The *lab technician* and the *specialist agents* **return** results of tests.
20. The *user* studies the results, and **posits** a diagnosis of achalasia.
21. The *tutor* **accepts** this diagnosis, once again, as in 18 above, displaying the preconditions for this action.
22. The *user* then **prescribes** BoTox as a treatment
23. The *tutor* **permits** this action (its precondition being a definitive diagnosis of achalasia).
24. The *user* decides he has had enough tutoring and **turns off** the tutor. The tutor's "unspoken" reaction to all the moves in the scenario can, however, still be seen by the user following the scenario in a log generated by the system.
25. The *user* **schedules** a visit in a month (month 33) for a follow-up.
26. At that time, the *VP instance* is **asymptomatic**, meaning that the BoTox was successful. However, BoTox's effects are only temporary, so there will be regression.
27. The next **follow-up** is in 9 months (month 42). The *user* **interviews** the patient who **responds** that she has significant difficulty swallowing.
28. The *user* **orders** the surgical procedure Heller myotomy, which cuts the lower esophageal sphincter to permit food to pass.
29. Three months later (month 45) is the next **follow-up**: the *user* asks about swallowing and also heartburn (since the hypotensive sphincter that results from Heller myotomy often leads to GERD). The *VP instance* **responds** that swallowing is fine but heartburn has developed.

In the continuation of the scenario, the user should treat the heartburn. Depending on this patient's inherent predispositions, her achalasia might regress, meaning that her lower esophageal sphincter would become tight again, reversing the GERD but reintroducing difficulty swallowing; alternatively, the achalasia might not return, meaning that the patient will have GERD for the rest of her life.

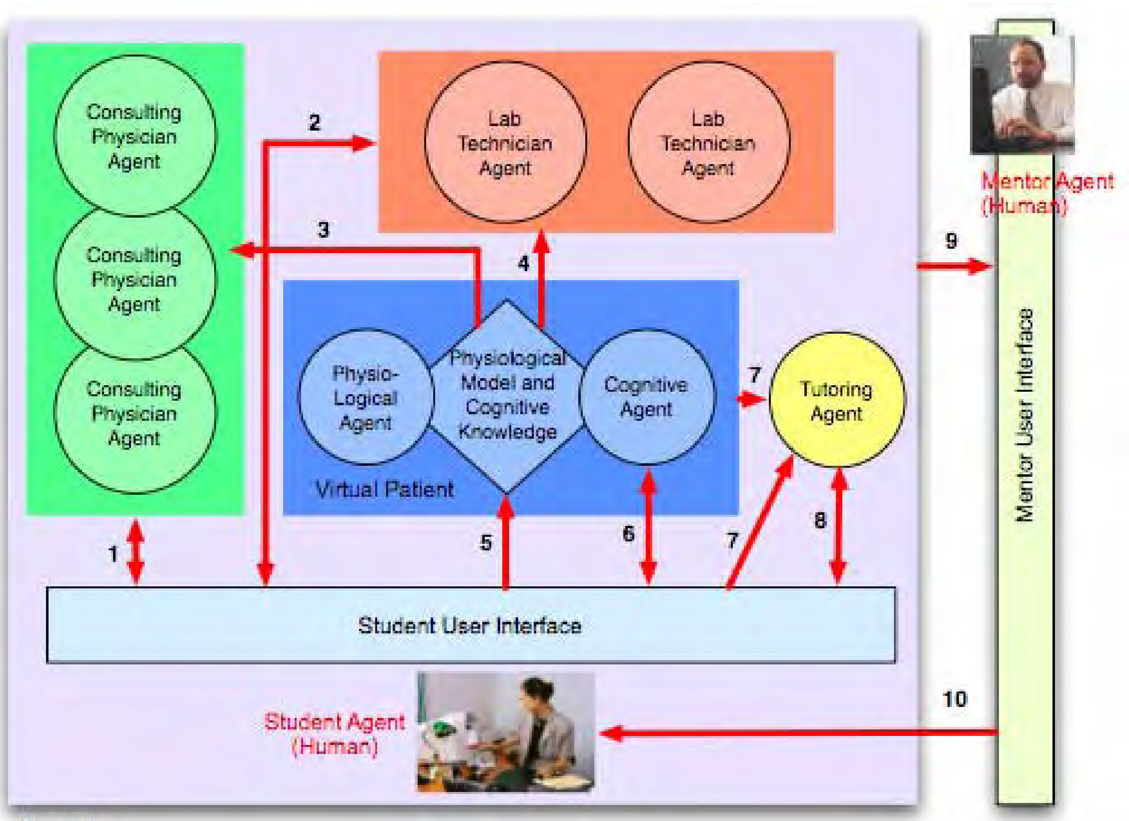
As this example illustrates, one of the novel aspects of the MVP system is how open-ended the scenarios are. An informal run-through with third-year medical students confirmed our assumption that the lack of a rigid structure – typically imposed by multiple-choice type learning environments – was both a surprise and a boon for students, since it so much more closely mimics what they will be required to do in clinical practice.

The Agent Network

A variety of views co-exist on what constitutes an intelligent agent. We divide our agents into high-level and low-level as follows.

High-level agents include actual humans, simulated humans (specialist consultants, lab technicians, the virtual tutor), and the physiological and cognitive sides of the virtual patient, which we choose to model as separate high-level agents. High-level agents are viewed as complex objects that can include models of the world and of self, including character traits, goals and plans. They also include various processing capabilities, such as decision making, language understanding and modifying a physiological profile. Figure 3 presents the high-level agents in the MVP network and the types of communication among them.

Low-level agents are, for us, not objects but processes. They may represent the specific capabilities of various high-level agents (e.g., language generation) or events in the world external to the high-level agents (e.g., exposure of a virtual patient to an environmental toxin). Low-level agents can be used to simulate individual physiological processes, both normal and pathological (i.e., diseases). Low-level agents realize a broad range of entities, from surgeries that affect the VP's physiology and anatomy, to external stressors that affect its cognitive and physical state, to individual physiological and pathological processes, to system functionalities that support simulation and tutoring goals (the patient creation agent, the information visualization agent, etc.).



Legend

1 From student: requests for information; from consultants: advice
2 From student: requests for labwork; from lab technicians: test results, augmented by interpretation by specialist
3 Consulting physicians get data from the physiological model of the virtual patient
4 Lab technicians get data from the physiological model of the virtual patient
5 Student can administer treatment, simulated by changing the physiological (and/or anatomical) model of the patient
6 From student: questions about symptoms etc.; from cognitive agent: responses in language
7 The tutoring agent collects information about the diagnostic and treatment processes
8 From the tutoring agent: advice, warnings, answers to user questions; from student: questions, responses to queries
9 The mentor observes the diagnostic and treatment as well as automatic tutoring processes
10 The mentor communicates directly with the student

Figure 3. The Maryland Virtual Patient (MVP) As Member of Its Multi-Agent Team that Includes Human and Simulated Agents

In addition to being categorized as high-level or low-level, the agents in the MVP network can also be characterized as deterministic or cognitive. In our approach, deterministic agents model physiological, environmental, chemical, etc., processes, while cognitive agents model sentient human behavior. Deterministic agents do not possess models of desires and intentions, they cannot be aware of other agents around them, cannot negotiate, cannot delegate and cannot have their decisions be influenced by purely agent-internal causes. For example, a disease agent can only advance the progression of the disease in accordance with the changing values of specific physiological properties of the virtual patient and the passage of time. Of the traditional triad of factors involved in modeling agency – beliefs, desires and intentions (see, e.g., Rao and Georgeff 1991) –

deterministic agents can be said to possess only the first, since it is plausible to interpret the knowledge of property values as the beliefs of an agent at a point in time.

The only MVP agent that is both high-level and deterministic is the physiological side of the VP. In our chosen application, there is no reason to model the physiological sides of the other simulated high-level agents in the system – the lab technicians, the consultants or the tutor. Note that there is a channel of communication between the physiological and the cognitive side of the virtual patient agent. It consists of the cognitive agent being able to trigger physiological processes and to receive from the physiological agent sensory information about symptoms (e.g., pain or difficulty swallowing). For example, swallowing is a deliberate conscious act, but only in its first stage: after the voluntary act of swallowing has been initiated (the gulp), the rest of the processes needed to get the bolus from the mouth to the stomach are deterministic physiological – though not pathological! – processes.

Two special agents in the MVP environment, the scheduler and the executor, support the operation and interaction of other agents.

In the subsections to follow we briefly describe the core agents in the MVP network, the data that supports their functioning, and their interactions with other agents in the network.

The virtual patient is the most complex agent in the system. This is because we model both its body and its mind. In other words, the virtual patient is for us a “double” physiological and cognitive agent. The physiological agent is a simulation of physiological and pathological processes. The cognitive agent combines the capabilities of perception (specifically, proprioception and language understanding), goal- and plan-based reasoning and action (decision-making and language generation).

The operation of the physiological agent is not directly controlled, though it can be influenced, by decisions and choices of the cognitive agent. Examples of such influences are lifestyle preferences, like smoking, regular exercise, and diet. The operation of the cognitive agent can, in turn, be influenced by the physiological agent. Indeed, the cognitive agent’s choice of goals, and the choice of plans for their attainment, will be influenced by the physiological agent’s physical state (e.g., disease, fatigue) and mental state (e.g., stress).

The physiological agent models the body as a collection of anatomical objects and physiological processes, including both normal and pathological ones (diseases). Whenever possible, disease processes are modeled as causal chains of component events (and, implicitly, states). These causal chains are encoded as complex events – i.e., scripts – in the system’s static knowledge (specifically, in its underlying ontology). However, in many cases, medicine does not at this time possess sufficient knowledge about biochemical mechanisms of disease progression to allow for the construction of completely causal scripts. This means that disease scripts must often contain a combination causal chains and empirical knowledge about the progression of a disease -- what we refer to as clinically derived “bridges”. In the current implementation, when it is not possible to encode a causal chain, the progression of the disease is divided into clinically relevant conceptual stages, and a set of value ranges of relevant physiological properties is encoded for the beginning and end of each stage. During simulation, values for interim time points are established through interpolation.

In our knowledge acquisition work we have found that expert clinicians like to express their opinions in terms of probabilities, e.g., “X% of patients develop stage N of disease D within M months of inception.” Though the use of value ranges instead of single values reflects this state of affairs, in our current application system the probabilities do not actually play a central role. This is because the system is supposed to train future physicians, and this is best done when the instructor can create an inventory of virtual patients carrying a particular disease such that the patients display the full spectrum of disease manifestations, symptom profiles and responses to treatments, not only the most common ones. Presenting the student with a carefully crafted set of such patient instances permits all of the necessary learning points to be targeted without the need to spend years of real-world training waiting for the less common patients to present. A case in point: in the evaluation of the SHERLOCK II system, which teaches electronics troubleshooting, it was reported that technicians learned more from using this system for 24 hours than from 4 years of work in the field (Evens and Michael 2006).

Once the progression of a disease reaches the symptomatic (i.e., clinical) stage, the simulation reaches the cognitive side of the double agent. Through proprioceptive perception, the cognitive agent becomes aware of symptoms such as pain or difficulty swallowing. Note that the experiencing of symptoms varies widely across patients and, accordingly, cannot be directly linked to given physiological states. However, a fixed inventory of symptoms is associated with each disease and expected ranges of values for each symptom can be asserted for each stage of the disease.

While the physiological agent has been built as a result of the formal encoding of expert knowledge about the progression of diseases and response to treatments, the knowledge encoded for the use of the tutor agent concentrates on formalizing the what expert physicians understand to be best clinical practices – for example, what preconditions should occur for particular physician actions to be warranted and what the best ways of scheduling these actions are. Our work on the tutoring agent to-date has concentrated on these contentful issues much more than on the specific pedagogical choices (so-called “tutoring moves”) that are at the center of interest in many intelligent tutoring projects in medicine and other areas (e.g., Evens and Michael 2006). At present, we implement only a small subset of choices available to a tutor. For example, different tutor settings allow more or less frequent interventions, the presence or absence of explanations for why certain suggested user actions were blocked, and so on. While we will continue to enhance the repertoire of choices in tutoring, our main goals are the adequate simulation of human physiology, the complete encoding of relevant best clinical practices, and the support of realistic natural language dialog between the user and the virtual patient.

The Physiological Agent of the VP

The VP physiology agent is modeled as a set of interconnected ontological objects representing human anatomy. Each object is described by a set of ontological properties and their associated value sets. Crucial among the properties are those that link the objects to typical events in which they participate. These events are usually complex – that is, include other, possibly also complex, events as their components. We call these complex events scripts, an example of which is swallowing. The swallow script involves numerous anatomical objects in the VP carrying out various roles. For example, the esophageal segments that comprise the esophagus are composed of various types of tissue, muscle and nerves; the segments act as *instruments* of the peristalsis that forces the swallowed bolus toward the stomach.

All events within the simulation environment are interpreted as low-level agents. Swallowing, as such an agent, knows what role each property of each component part of the esophagus plays in the process of swallowing. Specifically, it knows what property values are normal and what effects to post after a normal swallow, and it knows what property values are abnormal and what effects to post after an abnormal swallow, depending on which abnormality was encountered. For example, a tumor in an esophageal segment decreases the size of the esophageal lumen and thus causes the symptom "difficulty swallowing" in the patient; the larger the tumor, the greater the difficulty swallowing, until the point where swallowing is blocked altogether. For the task of modeling esophageal diseases, having a normal, working model of the process of swallowing is useful, since asking a patient to swallow, and asking whether that triggered any symptoms, is a realistic component of a simulated office visit.

At first blush, it might seem preferable to have a maximally complete model of normal human anatomy and physiology before progressing to disease modeling, but we have found this not to be the case for three reasons:

1. Creating formal models of everything known about human physiology would require an unsupportable amount of time and resources.
2. Even if such models could be created, they would represent a grain size of description not needed for our applications.
3. Many of the processes of human physiology – both normal and pathological – are not understood by the medical community, meaning that modeling must anyway combine aspects of known causal chains and clinical observations that we call "bridges."

In short, all modeling in the MVP system is task-oriented, with both normal and pathological processes being modeled on an "as needed" basis. Achieving a useful balance between causal chains, bridges, and grain size could be considered the art of application-oriented modeling.

At any given time, the model of the normal human contains whatever normal anatomical and physiological knowledge was compiled to cover the diseases currently available in the system. So, although at present our virtual humans do not have a highly developed model of the circulatory system, as soon as we have completed the circulatory model – which is currently under development to support the modeling of heart disease – all virtual humans will be endowed with all the associated functionalities and property values.

Disease Agents

In the MVP system, diseases are modeled as processes (low-level agents) that cause changes in key property values of a patient over time. For each disease, a set number of conceptual stages is established and typical values or ranges of values for each property are associated with each stage. Relevant property values at the start or end of each stage are recorded explicitly, while values for times between stage boundaries are interpolated; the interpolation currently uses a linear function, though other functions could as easily be employed.

A disease model includes a combination of fixed and variable features. For example, although the number of stages for a given disease is fixed, the duration of each stage is variable. Similarly, although the values for some physiological properties undergo fixed changes across patients, the values for other physiological properties are variable across patients, within a specified range. The combination of fixed and variable features represents, we believe, the golden mean for disease modeling. On the one hand, each disease model is sufficiently constrained so that patients suffering from the disease must show appropriate physiological manifestations of it. On the other hand, each disease model is sufficiently flexible to permit individual patients to differ in clinically relevant ways, as selected by patient authors.

For illustration, we take a simple disease model, the one for scleroderma esophagus. It is simple because the causal chains driving the disease are not known to medicine, making it necessary to model the disease purely in terms of clinical observations over time.[10] Table 1 shows the physiological properties that change over time. The values in each cell represent the values at the beginning of each stage: so a given patient might have basal lower esophageal sphincter (LES) pressure of 18 mmHg at the start of t0, 11 mmHg at the start of t1, etc. Legal value ranges are indicated, with defaults in parentheses. Properties for which no choice of range is provided (e.g., level of anti-nuclear antibodies) are fixed across all patients (subject matter experts saw no benefit to making such properties variable across patients). The symptom profile for patients with scleroderma esophagus is shown in Table 2, with value ranges and defaults represented using the same conventions.

Table 1. Physiological properties that change due to scleroderma esophagus

	t0	**t1**	**t2**	**t3**	**t4**
Level of anti-nuclear antibodies	0	1	2	3	4
Peristalsis efficacy	normal peristalsis	intermittent-peristalsis	aperistalsis	aperistalsis	aperistalsis
Basal LES pressure (in mmHg)	0-25 (15)	0-15 (10)	0-10 (5)	0-10 (3)	0
Erythrocyte sedimentation rate	0-15 (12)	5-20 (10)	20-30 (25)	31-40 (35)	41-50 (45)
Stage duration (in years)	1-5 (1)	1-5 (1)	1-5 (1)	1-5 (1)	1-5 (1)

Table 2. Symptom profile for patients with scleroderma esophagus

	t0	**t1**	**t2**	**t3**	**t4**
Raynaud's symptoms	no	yes	yes	yes	yes

[10] We choose a simple disease model here in order not to stray from the main thrust of the article: the agent network in the system. For descriptions of more complex disease models, see McShane *forthcoming* (for achalasia) and McShane *submitted* (for GERD).

Skin tightness	0	0-.3 (.1)	.1-.4 (.3)	.3-.6 (.4)	.5-.8 (.7)

Scleroderma esophagus is a disease whose path cannot be altered by interventions by any outside agents. However, the physiological changes it causes give rise to another disease: GERD. Specifically, when the basal LES pressure drops below 10 mmHg (typically in the t2 or t3 stage of scleroderma esophagus), the amount of stomach reflux permitted by the hypotensive sphincter will be sufficient for irritating processes of the esophageal lining to begin – i.e., GERD. There is nothing in the disease model for scleroderma esophagus about GERD because there need not be: the initiation of GERD is handled by an object-oriented condition on the LES that is triggered when the LES pressure drops below 10 mmHg.

In the simple scleroderma esophagus model, there are few parameterizable values; however this is not so for all diseases. For example, GERD has six radically different clinical manifestations – from harmless but symptom-inducing irritation of the esophageal lining to esophageal cancer. Which path a patient takes depends upon a number of parameters, including genetic predispositions, lifestyle habits, and certain physiological features, like LES pressure, that can be affected by outside agents.

The Cognitive Agent of the VP

As mentioned above, some diseases, like scleroderma esophagus, cannot be affected by an external agents, but many others can. The cognitive agent can affect disease progression by choosing **lifestyle habits** and by **compliance or non-compliance with medication regimens.** Let us consider the example of GERD, one of whose causes is a hypotensive LES.[11] The LES can be made more hypotensive by lifestyle habits like consuming caffeine and eating chocolate. The more hypotensive the LES, the higher the daily acid exposure of the esophageal lining and the faster the disease progression, based on the causal chain of events used to model GERD (see McShane et al. submitted).

Consider a patient whom we'll call Patient A. This patient has a basal LES pressure of 4 mmHg, he engages in the GERD-irritating habit of consuming caffeine, and he is genetically predisposed to a form of GERD that progresses to the level of producing esophageal erosions but never produces ulcers, cancer or other complications. If the patient does nothing to improve his disease course, it will proceed as in Figure 4, which shows just four of the dozen or so property values relevant for GERD.[12] However, if Patient B stops his caffeine habit in month 5 of the disease, when his heartburn is already at a .5 on the scale of 0-1, his LES pressure will increase slightly, decreasing acid exposure and producing a disease course as in Figure 5, where heartburn severity reaches

[11] The other cause is transient relaxations of the LES, which, like a hypotensive LES, permit excessive exposure of the esophageal lining to acidic stomach contents.

[12] To briefly explain the properties shown: when preclinical esophageal irritation reaches its maximum, the clinical (i.e., symptom-producing) stage of the disease begins, in this case, causing heartburn; when the mucosa depth reaches 0, erosive esophagitis begins (in the simulation an erosion object would be instantiated); and when the mucosa depth reaches 0, the erosion has reached its maximum depth – any further eroding would constitute an ulcer, which this patient is not predisposed to get. In this and subsequent figures, when different property values have different scales of measurement, we normalize them to make their changes over time most visible in the graph. We call the resulting scale an abstract scale.

its maximum at around month 10 rather than month 9. Clearly, for this patient, lifestyle modification is not enough to stop or reverse the disease course, but it causes some modification of disease path.

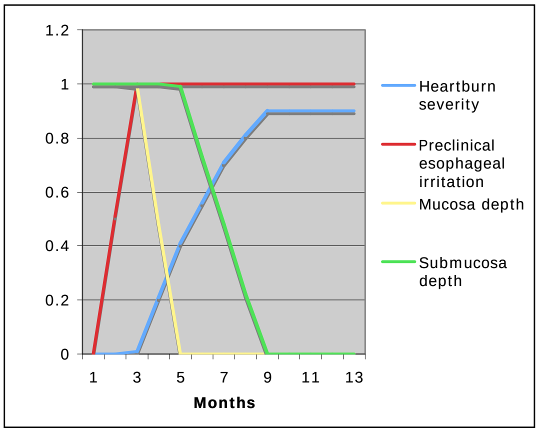


Figure 4. Patient A with no interventions.

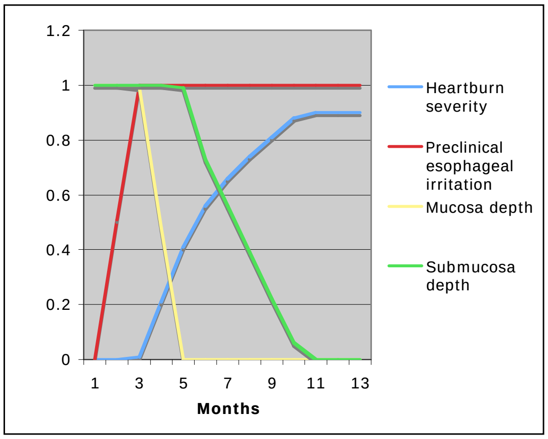


Figure 5. Patient A with lifestyle modifications in month 5.

Now assume that in month 5 the patient decides to see a doctor and is put on medication that, for him, is effective. (See Figure 6.) Assume further that he takes that medication for three months then stops taking it. During the period when he is taking it, his disease heals and remains healed, but as soon as he stops taking it, the disease begins to progress again, first asymptomatically (in the preclinical phase), then symptomatically.

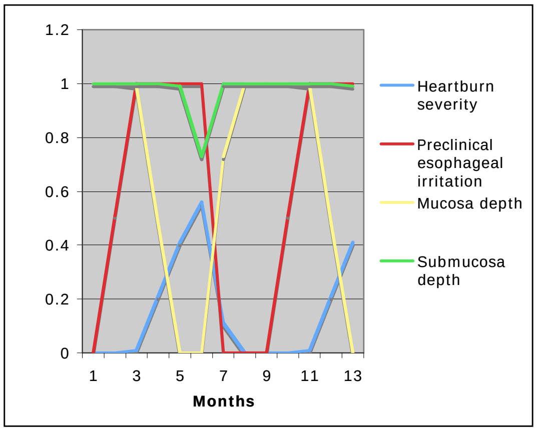


Figure 6. Patient A with medication starting in month 5, ending in month 8.

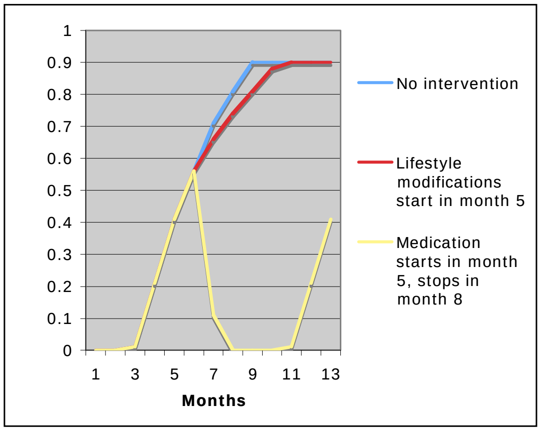


Figure 7. Patient A: heartburn severity under different treatment scenarios.

Figure 7 compares the symptom-related experience of the cognitive agent of Patient A under these three scenarios using just the symptom "heartburn severity".

The behavior of the cognitive agent with respect to these choices is determined by an inventory of personality traits that can be selected by the patient author. Our initial inventory, which has been sufficient for esophageal diseases, includes pain threshold, willingness to seek medical help and compliance with medication regimens. However, the group of diseases we are currently in the process of modeling – cardiovascular diseases – relies much more heavily on personality and lifestyle factors, which will require an increase in the inventory of ways in which the cognitive agent can influence the operation of the physiological agent.

The above discussion relates to only those components of the VP cognitive agent that model influences on the VP physiological agent. Of course, the cognitive agent is also responsible for a wide variety of sensory and proprioceptive perception, reasoning, decision making and inter-agent communication processes. Figure 9 illustrates the component processors and static knowledge resources of the VP cognitive agent. The figure includes both stable components that have largely been developed and components under development at the time of writing.

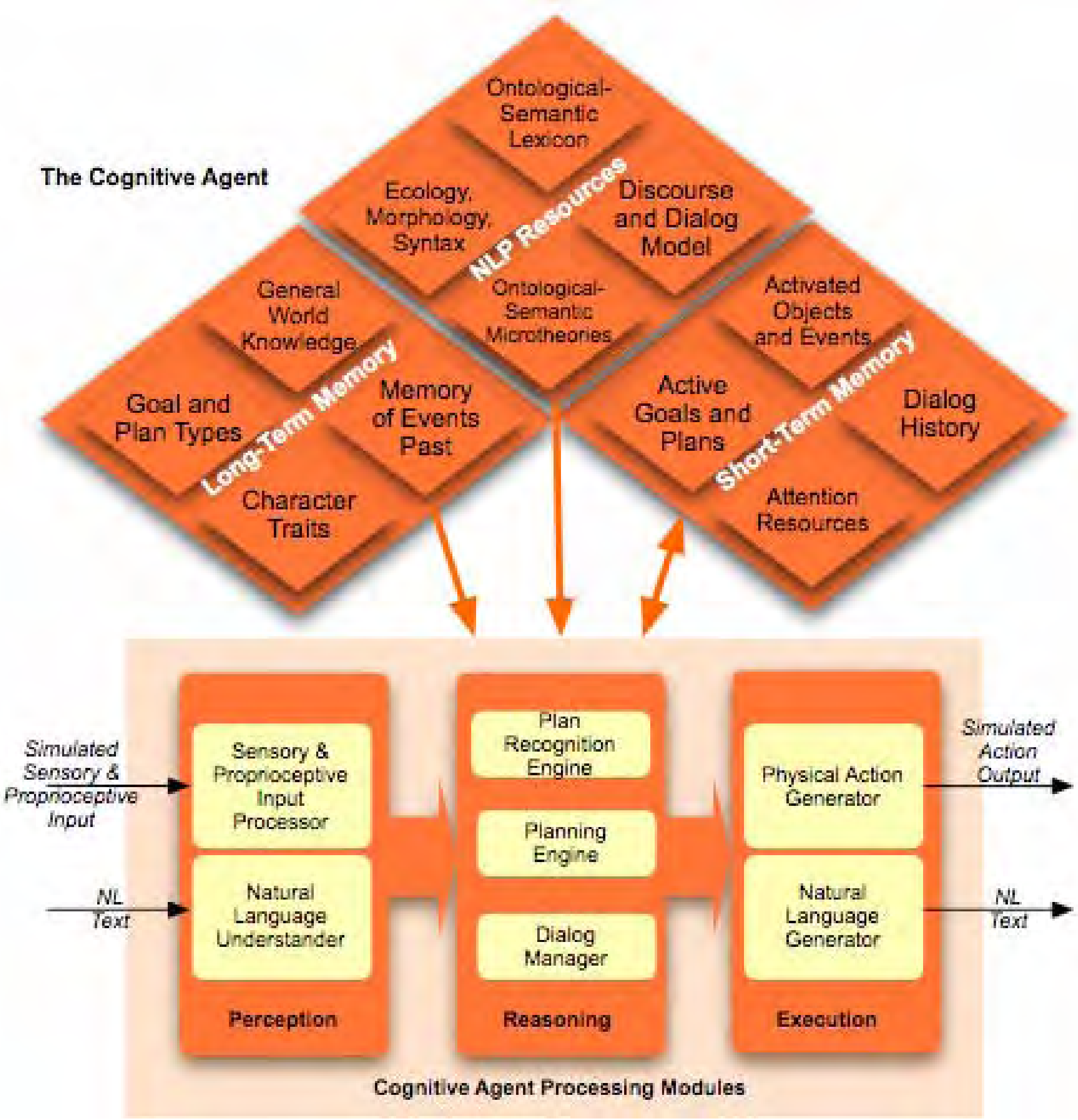

The Cognitive Agent
Ontological-Semantic Lexicon
Ecology, Morphology, Syntax
NLP Resources
Discourse and Dialog Model
Ontological-Semantic Microtheories
General World Knowledge
Goal and Plan Types
Long-Term Memory
Memory of Events Past
Character Traits
Activated Objects and Events
Active Goals and Plans
Short-Term Memory
Dialog History
Attention Resources
Simulated Sensory & Proprioceptive Input
NL Text
Sensory & Proprioceptive Input Processor
Natural Language Understander
Perception
Plan Recognition Engine
Planning Engine
Dialog Manager
Reasoning
Physical Action Generator
Natural Language Generator
Execution
Simulated Action Output
NL Text
Cognitive Agent Processing Modules

Figure 9. The Processors and the Static Knowledge Modules of the Cognitive Agent component of the virtual patient. Work is ongoing on all these modules. The constraints of the application help to keep the inventory of goal and plan types, the character traits and the attention resources relatively small. The general world knowledge (the agent's ontology) and its lexicon are close to being complete. The discourse and dialog processing is a core direction of work in 2007-8.

The User

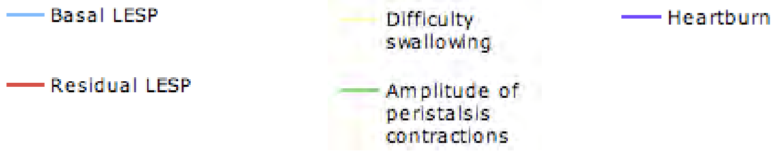
At the beginning of a simulation session, the system presents the user with a virtual patient about whose diagnosis he initially has no knowledge. The user then attempts to manage the patient by conducting office interviews, ordering diagnostic tests and prescribing treatments. The basic strategies of patient management were illustrated in the example in Section 2.2.

Answers to user questions and results of tests are stored in the user's copy of the patient profile, represented as a patient chart (see left pane of Figure 1). At the beginning of the session, the chart is empty and the user's cognitive model of the patient is generic – it is just a model of the generalized human. The process of diagnosis results in a gradual modification of the user's copy of the patient's profile so that in the case of successful diagnosis, it closely resembles the actual physiological model of the patient, at least, with respect to the properties relevant to the patient's complaint. A good analog to this process of gradual uncovering of the user profile is the game of Battleship, where the players gradually determine the positions of their opponent's ships on a grid.

At any point during the management of the patient, the user may prescribe treatments.[13] In other words, the system allows the user not only to issue queries but also to intervene in the simulation, changing property values within the patient. Any single change can induce other changes – that is, the operation of an agent can at any time activate the operation of another agent.

Consider the example of Patient B, who suffers from achalasia, a disease that increases the basal and residual pressure of the LES, making swallowing increasingly difficult. The four graphs below show different outcomes based on different interventions by the user.

Legend for Figures 9-12:



[13] In this discussion, we disregard the tutor agent that can, in some pedagogical configurations, prevent a user action.

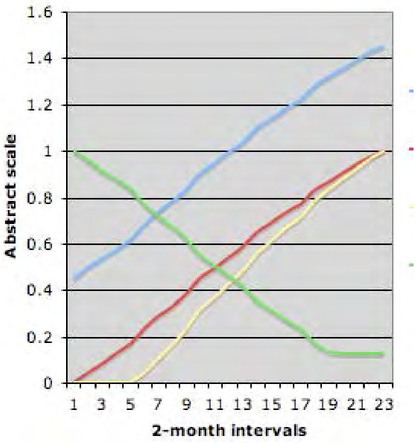


Figure 9. Patient B with no interventions.

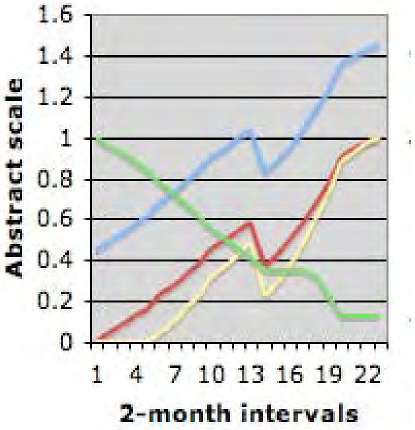


Figure 10. Patient B with BoToX in month 26.

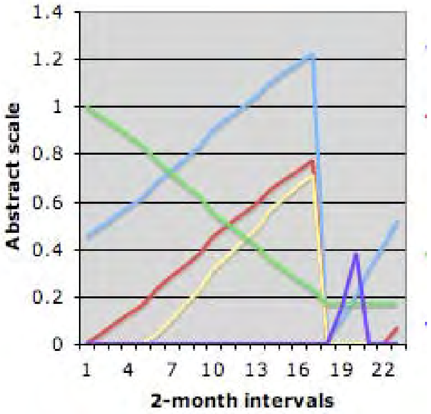


Figure 11. Patient B with Heller myotomy in month 34.

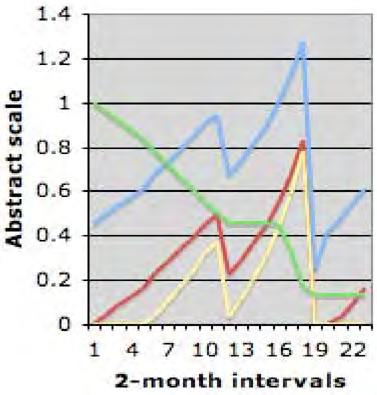


Figure 12. Patient B with BoTox in month 22 and pneumatic dilation in month 36.

Figure 9 shows the disease progression with no intervention. Figure 10 shows what happens when BoTox is administered: this treatment, if effective for the given patient, lowers basal LES pressure temporarily, but as it wears off the disease path returns to where it would have been if the treatment had not been given. Figure 11 shows intervention with the surgical procedure Heller myotomy in month 34. This procedure cuts the LES, typically leaving it with a pressure of near 0. With such a low LES pressure, a patient generally experiences excessive reflux and begins to suffer from GERD, as shown by the spike in heartburn around month 38. The reason the heartburn does not continue indefinitely is that the author of this patient instance decided that Heller myotomy would not provide definitive treatment for this particular patient; as a result, the tightening of the LES that is characteristic of achalasia again takes hold over time. By month 44 the patient's LES pressure has increased sufficiently to stop the GERD symptoms and, if we were to track the disease path even longer, we would see that his difficulty swallowing would continue to increase. Figure 12 shows two interventions: first, BoTox is administered and provides a temporary decrease in symptoms; then, when symptoms return to a high level, the endoscopic procedure pneumatic dilation is carried out, which tears the LES using an inflated balloon. This procedure, like the Heller myotomy, can be definitive or not definitive, depending on the patient's predispositions.

The four scenarios above illustrate but a handful of the thousands of scenarios one could create using Patient B, since any of the treatments could be administered in any combination at any time. In addition, there are *incorrect* treatments that could be administered which might be harmless or might worsen the patient's condition, creating a more complicated case that the user must manage. Moreover, in addition to Patient B, who has a specific inventory of disease-related property values, hundreds of other patients with this disease could be authored, making the scope of cases truly wide and differentiated.

Virtual Medical Personnel

Currently, the agents simulating medical personnel in the system include lab technicians, specialists that interpret test results and specialists that carry out procedures. In the future, we will include virtual user assistants to carry out physical exams.

Virtual lab technicians are agents that know which property values each test targets and check those values in the current state of the virtual patient, returning the values as strings. Test-interpretation specialists are endowed with a data set that associates given property values and combinations of property values with given interpretations (strings).

Specialists that carry out procedures are currently modeled in a rather simplified way: the results of the various tests and procedures, if carried out on the patient at various times, are pre-recorded and those results, in a sense, encapsulate the work of the specialist. In future, we will endow the specialist agents with a sufficient number of features to permit the autonomous computation of the quality of their results.

Environmental Agents

Environmental agents are a diverse set of low-level agents external to the VP that can have a profound effect on virtual patients. We expect many of these to be launched randomly within parameters set by the patient author. Two examples of environmental agents are stressors and the exposure of the VP to entities, like viruses and toxins that can cause disease processes. For example, for patients with heart disease, both long-term and acute stressors significantly affect disease course. To model the influence of environmental agents like stressors, the MVP environment will initiate, at points in time prescribed by the VP instance author, the operation of agents that raise the VP's stress levels (e.g., a car accident, divorce). Although participants in some such events could, in principle, be modeled as high-level agents (e.g., the driver of the car that caused that accident or the microorganism whose activities caused a disease), there is no practical reason for doing so in our applications. Instead, all environmental agents are interpreted as low-level agents.

The Tutor Agent (The Virtual Mentor)

The tutor agent, the virtual mentor, is one of two pedagogical agents in the environment, the other being a human mentor whose potential participation we will not detail here.

Much of the knowledge of the tutor agent derives from what we call “preconditions for good clinical practice”. These are formally specified conditions that should be satisfied before a given hypothesis or diagnosis is posited, or before a given test or treatment is launched. For example, before carrying out a Heller myotomy, a definitive diagnosis of achalasia must be made; and in order for a definitive diagnosis of achalasia to be made, the following preconditions must hold:

ALL OF:

1. ONE OF:
   - bird’s beak ; one interpretation generated from barium swallow test
   - LESP > 45 ; basal LES pressure
2. aperistalsis
3. ONE OF:
   - dysphagia ; difficulty swallowing
   - regurgitation

So, if the tutor agent is enabled and the user attempts to perform a Heller myotomy without fulfilling the necessary preconditions, the tutor will generate a message. The nature of the message depends upon the tutoring option selected: at the highly informative end of the scale, the tutor can provide all fulfilled and unfulfilled preconditions, with the former being in green and the latter being in red; at the most uninformative end of the scale, the tutor can respond “Invalid action”, leaving the user to figure out what needs to be done to fill the lacunae.

Certain tutor settings make it impossible for the user to carry out “bad practice” actions. In this way, the tutor not only affects the human user, it also affects the virtual patient run by making impossible certain paths of disease progression and healing. Consider again the achalasia patient we called Patient B. With the tutoring disabled, the user is permitted to treat the patient with BoTox in month 18, but with tutoring enabled, the first time the user could administer BoTox would be in month 26, which is the earliest time that all the preconditions for administering it are fulfilled.

Another way in which the tutor will, in the future, be able to affect both the user and the virtual patient is by suggesting what to do next when the user asks for help. This functionality requires that the tutor be endowed with more and different knowledge than that needed for the current preconditions-driven tutoring method. Specifically, it will need to decide – based on its knowledge of the good practices for diagnosis and treatment – which of many moves available at any given time in the process is optimal and be able to convey the reasoning behind that decision to the user.

Much of the work on intelligent tutoring systems concentrates on the implementation of a variety of pedagogical “moves” (Evens and Michael 2006). We have already implemented a subset of such moves by introducing different levels of tutor intervention, and will integrate more as work proceeds.

An automatic tutor can perform a variety of useful tasks. For example, it can track the decisions the human user takes and suggest alternative (better) courses of action, alert the user to errors or unfulfilled preconditions for various actions, and so on. One of the most important roles of a tutor in a clinical setting is providing the human user with advice about *what to do next*. In order for the virtual tutor to decide what should be done next, it must combine context-independent knowledge with context-specific reasoning.

The context-independent knowledge of our virtual tutor covers, non-exhaustively:

1. **high-level best clinical practices**: e.g., that the physician should interview the patient before all else (in non-critical situations); that he should posit a working hypothesis or diagnosis before ordering tests and procedures; that each test and procedure should be ordered only if the patient meets certain objective criteria warranting it, etc.;
2. **diseases**: their signs and symptoms over time; the chief complaints they give rise to; other diseases with overlapping signs and symptoms; variations in disease manifestation across patients; what constitutes sufficient evidence to hypothesize, clinically diagnose or definitively diagnose each disease;
3. **diagnostic tests**: the specific criteria that should be met before ordering them; what they test for and what kinds of results they return; potential complications and their frequency; the availability and time frame of each test;
4. **interventions**: the specific criteria that should be met before ordering them; their projected outcome; potential side-effects and the frequency of those side-effects; the availability of qualified specialists to carry out various procedures.

Context-specific reasoning combines this static knowledge with knowledge about the patient and the student that the tutor compiles during the course of the given simulation run. Such dynamically created knowledge includes: (a) the current state and past history of the patient, as elicited by the student through patient interviews, (b) each of the past actions of the student (questions asked, tests ordered, interventions performed, hypotheses and diagnoses posited), and (c) the content of any past interactions between the tutor and the student (the student asking questions and the tutor answering; the tutor intervening to stop the student from making an ill-advised move, etc.). Note that the tutor does not have omniscient knowledge of the patient's physiology since no physician can work from that unrealistic starting point – it knows about the patient exactly what the student knows about the patient. The difference between the tutor and the student in this respect is that the tutor embodies the knowledge of best clinical procedures, as possessed by expert clinicians and encoded by knowledge engineers in the knowledge resources of the system.

The tutor can use its combined static knowledge and reasoning capabilities to function in a number of tutoring modalities, including:

1. providing step-by-step commentary about whether each move is clinically appropriate and why;
2. stopping the student before he carries out clinically inappropriate moves;

3. showing the student how to fulfill the unfulfilled preconditions for a given move;
4. suggesting which move(s) should be carried out next and why.

The first three of these tutoring modalities – as well as a number of variations on the theme – are available in the first release of MVP, and the last one is currently under development. Let us consider one scenario that highlights the adaptive nature of the tutor's reasoning capabilities:

1. The patient presents with the chief complaint "occasional difficulty swallowing."
2. The student interviews the patient, finding out that the only other symptom is occasional mild chest pain.
3. The student hypothesizes the disease 'achalasia.'
4. The student orders an esophagogastroduodenoscopy (EGD). Negative results for the EGD are returned by the lab technician and specialist agents.
5. The student then orders a barium swallow. It reveals a slight narrowing at junction of the stomach and the esophagus (the GE junction).
6. At this point the student asks what he should do next.

During the simulation, whether or not the step-by-step commentary function is enabled, the tutor evaluates each move the student makes, saving the evaluations in a log that can be reviewed later by the student and/or teacher. The tutor would approve of step 2 as long as sufficient questions about related diseases have been asked. For example, since the patient complains of chest pain, questions about other symptoms of heart disease should be asked, since chest pain is a primary symptom of heart disease. The inventory of symptoms that should be asked about is *dynamically generated* based on patient responses to each symptom question: the more symptoms the patient has, the more associated symptoms must be asked about, to rule out similarly presenting diseases.

The tutor would approve of step 3 but would suggest that a better hypothesis would be "motility disorder," which is a superclass of achalasia, since there is no evidence at this point in the proceedings suggesting which specific motility disorder this actually is. The tutor evaluates whether a hypothesis is reasonable on the basis of the filler of the ontological property SUFFICIENT-EVIDENCE-TO-HYPOTHESIZE, which is defined for each disease and class of diseases. If the above property has the value DIFFICULTY-SWALLOWING listed, this is sufficient grounds to hypothesize a MOTILITY-DISORDER, and is also sufficient grounds to hypothesize its child, ACHALASIA, but hypothesizing the more general disorder is better clinical practice in the absence of further evidence – a generalization known by the tutor.

The tutor would approve of step 4 on the basis of the clinical practice of *first ruling out any alarm signals* (i.e., potential immediate causes of danger). Difficulty swallowing can suggest a tumor, which could be cancerous, so the first action should be to rule that out. The need to rule out cancer in the presence of the symptom DIFFICULTY-SWALLOWING is recorded using the property TRIGGER-ALERT in the ontological description of DIFFICULTY-SWALLOWING, as follows:

```
DIFFICULTY-SWALLOWING
   TRIGGER-ALERT        TUMOR (LOCATION ESOPHAGUS)
```

Fillers of the property TRIGGER-ALERT should be pursued first, even if the related condition is unlikely and does not represent the current working hypothesis. Ontologically recorded knowledge about EGD includes the fact that it can detect tumors. It also includes the fact that DIFFICULTY-SWALLOWING, by itself, is a sufficient condition to order an EGD. As a result, the tutor determines that the student has acted correctly in ordering an EGD as the first study.

Whereas the EGD was ordered in step 4 in order to rule out a potential problem (tumor), the test ordered at step 5 is intended to *provide evidence to confirm the current hypothesis.* Ordering tests to confirm a hypothesis represents clinically correct behavior, since for many diseases (including achalasia) a diagnosis must be made before any treatment can be launched.

It is not surprising that the student would ask for help after the results of the barium swallow are received. He or she had a hypothesis but that hypothesis was not confirmed by the test ordered: the barium swallow would have to have shown a finding known as "bird's beak" at the GE junction – rather than just a slight narrowing – if achalasia were to be diagnosed. The student has not fulfilled the preconditions to order any more tests; and since a diagnosis of achalasia has not been confirmed, no treatments can be launched. *What should I do now?*, the student asks the tutor.

At this point, the tutor reviews the path taken so far and searches for any points at which other moves might have been taken. Formally, this means comparing the preconditions for all the available actions with the knowledge available at the time of each move. In this case, the student acted similarly to the way the tutor would have acted except for the following: (a) as described above, the tutor would have posited a motility disorder, not achalasia; and (b) the tutor might have sent the patient home after receiving the results of the EGD, since the patient was in no danger and its symptoms were very mild (whether to go ahead with the barium swallow or wait and see how the symptoms progress is a judgment call; an experienced diagnostician might have guessed that there would be insufficient evidence to diagnose any disease at this point, based on the patient interview). The tutor would point out to the student the slight deviations from how it, the tutor, would have handled the case, and would tell the student that at this time the best thing to do is to send the patient home, with a recall in a few months or if symptoms become significantly more pronounced.

This option of "wait and see" is one aspect of the system that both makes it extremely open-ended and trains students to do something that they are known to find uncomfortable – namely, not take immediate action. Most training environments for medical students tell students that they must do something immediately and give them a choice of what to do. However, physicians frequently must simply wait to see how a disease plays out, reassuring the patient and sending him home. In the scenario above, this is exactly the advice the tutor would give the student. The point, however, is that all of the tutor's decisions are based on comparing its static knowledge with the dynamically changing evidence available about the state of the patient and the knowledge of the student.

Architecture and Control Agents

This section presents an informal, content-oriented description of the operation of the MVP. The MVP system uses agenda-style control that operates on data stored in the fact repositories of each of the high-level agents.[14] Each high-level agent – i.e., virtual human – has its own blackboard, which represents its current inventory of property values. The agenda is implemented as a temporally ordered list of agenda slots. Each agenda slot is a set of event instances scheduled for execution at a particular time.[15] The frequency of agenda slots (the clock cycle period) can be manipulated in the MVP system: the temporal scale of the simulation can be changed in a broad range between milliseconds and years.

Simulated events (scripts) can be simple or complex. A complex event contains subevents whereas a simple, atomic event does not. Whether an event is modeled as simple or complex generally depends upon the selected grain-size of description: for example, in a non-medical application the event of swallowing might be modeled as a simple event, but in our medical application it includes dozens of subevents describing component physiological processes that can be affected in various ways by various disease processes.

From the standpoint of MVP operation, the most important properties of event instances are:

1. **Preconditions-effects**, whose values are effects of the event, each coupled with a set of its own preconditions. For example, for event E, if precondition A holds, then the event will be sent to the scheduler to be rescheduled in the next agenda slot (this is the way we model continuing events), whereas if precondition B holds, changes to values of specified properties will be posted in appropriate fact repositories. Preconditions typically involve one or more property values of the agent, of the world or the simulation process itself; the completion of another event is another typical type of precondition. Effects typically involve changing property values of the VP or scheduling new events.
2. **Subevents** are defined for complex events. Subevents can themselves be complex events, with no conceptual limit on the number of nesting levels. The subevents of an event are organized in a directed graph, whose nodes represent the subevents and whose edges represent a temporal ordering of the subevents. Whenever the system

[14] Each agent has its own fact repository that reflects its private knowledge (beliefs) about the world and includes a record of current and past events involving the agent or known to it. Thus, for example, the user's fact repository includes the subset of the VP physiology profile revealed to the user and the record of past events recorded in the patient chart. The complete VP physiology profile is the main fact repository in the system – all the changes to the VP properties are recorded there.

[15] For purposes of simulating parallelism, agenda slots have been implemented as last-in-first-out queues.

knowledge allows, the temporal precedence relations are also marked as causal connections, which subsume the temporal ordering.

3. **Duration**. For simple, atomic events, the stated duration is an important (though not the only) factor that determines the time at which its effects are posted by the executor. The duration of complex event instances is computed as the sum of the durations of all the subevents on a particular path in the subevent graph.

The control agents driving the MVP operation are the scheduler, the executor and a (potentially large) set of "demon" agents, which are procedural attachments to properties in the high-level agents that trigger events as a result of specified changes in the values of the properties to which they are attached. For example, if the pressure of a virtual patient's lower esophageal sphincter drops below 10 mmHg (as by a surgical intervention), this state of affairs is sensed by the corresponding demon, which triggers the action of scheduling the complex event "GERD" (and as a result of this, the VP physiological agent gets this disease). The MVP system, thus, is a mixed expectation- and data-driven system. The demons trigger reactive, data-driven events. Expectation-driven events (scripts and plans) are encoded in the ontology. The scheduler and the executor manipulate both kinds of events.

The **scheduler** performs the following operations:

1. Accepting requests for events to be put on the agenda (see below for the inventory of types of requests);
2. Placing those events in appropriate time slots on the agenda; and
3. Removing events from the agenda; removal of an event can be caused by (a) the executor having determined that the preconditions of none of its effects are fulfilled; or (b) a direct request from an agent (e.g., the patient cancels a follow-up visit).

Any rescheduling operation – e.g., moving an appointment to a different date – is interpreted as a removal action followed by a scheduling action.

The **executor** performs the following operations for each event it processes:

1. Checking the preconditions of each of the effects of an event, and
2. Posting each of the effects whose preconditions hold.

One of the possible effects of an event is sending a request to the scheduler to (re)schedule it. The precondition associated with this effect involves the stated duration of the event and, optionally, property values of the patient or the world at the time when the precondition is evaluated.

The latter constraints are included to alleviate the frame problem – to account for potential changes to the world caused by other agents (events). Theoretically, the rescheduling should occur for the next agenda slot, and the rescheduling process should repeat until the agenda reaches the slot corresponding to the end of the duration of the event (unless external influences expressed in additional preconditions dictate otherwise). Considering that the clock cycle period in the MVP can be on the order of milliseconds, this will lead to inefficiencies. Therefore, in practice, we assign a default clock cycle period to each kind of event (e.g., one second for esophageal peristalsis; one month for a stage in the GERD event, etc.) and specify it in the rescheduling request. This, in turn, leads to constraints on the possible times of occurrence of external events: for example, the author's suggestion that a stressor be scheduled for a given patient at time T might be "relaxed" by the scheduler, such that it is scheduled at T +/- some amount, such that it does not occur between planned agenda slots. As yet, such efficiency-oriented constraints have not been shown to affect the realism of the simulation or to limit the pedagogical impact of the system.

Requests for putting events on the agenda can be sent to the scheduler in several ways:

1. The scheduler can have a "standing request" for scheduling normal physiological events (breathing, heartbeat, etc.) at appropriate intervals. One of the effects for such events will be to schedule another instance of this event once the previous instance finishes executing.
2. The VP instance author may stipulate that certain events should be scheduled at specific times. For example, the author requests the scheduling of disease scripts (agents) and environmental agents (e.g., exposure to a toxin) for appropriate times.
3. User interventions – the asking of questions, the ordering of tests or procedures, the scheduling of a VP's follow-up visit – are requests to the scheduler that those events be put on the agenda.
4. Actions by the VP cognitive agent, like ceasing to take some medication, are sent to the scheduler.
5. Effects of actions being executed by the executor may include sending a scheduling request. For example, when the event of the user asking the VP agent a question is executed, the effect is scheduling the events of the VP agent understanding this question and formulating the answer to it. Another example is scheduling the next subevent in a complex event once the previous event has completed its execution.
6. Requests for scheduling events can appear as a result of the operation of demons.

Medical Knowledge Creation and Formalization

The core medical knowledge in the system covers human physiology, pathology, relevant farmacology and a record of physician expertise – the "best clinical practices." This knowledge is needed for the support of simulating disease progression, disease interaction and medical interventions as well as for the support of the intelligent tutor agent in its task of helping the user train in the best clinical practices.

The MVP project places significant demands on physician-informants to render complex, multi-scale knowledge in a form that can be implemented computationally – naturally, with a knowledge engineer mediating between physicians and programmers. Physicians must distill their extensive and tightly coupled physiological and clinical knowledge into the most relevant subset, and express it in the most concrete terms. Not infrequently, they are also called upon to hypothesize about the unknowable, like the state of a patient experiencing a pre-clinical stage of disease, or the state of a patient after an effective treatment that is never, in real life, followed up by objective tests. Such hypotheses reflect the mental models of specific experts, which might differ in subtle ways from those of other experts. However, such differences, we would suggest, have little bearing on the ultimate goal of this specific project: to create MVPs whose behavior is sufficiently life-like to further specific educational goals.

The process of extracting knowledge from the experts and from other sources – medical literature, existing repositories of structured medical knowledge – is at present not automated and requires the participation of a knowledge engineer. We have developed support tools for this task, but we do not expect to be able to automate this task any time soon. Knowledge models extracted from the experts and other sources are formulated in the metalanguage used by the OntoSem environment, specifically, its ontology.

The Ontology

The term "ontology" has been used of late to refer to a heterogeneous group of entities, so we will define our use of the term by referring to our ontology, the OntoSem ontology (stemming from the theory of Ontological Semantics; Nirenburg and Raskin 2004). The OntoSem ontology is fundamentally different from most other "ontologies" in its emphasis on rich property-based descriptions that are not present in the many hierarchical trees of words or concepts available both within the medical domain (e.g., UMLS; Bodenreider 2004) and outside of it (e.g., WordNet; http://wordnet.princeton.edu). One ontological model that does contain useful properties is the Foundational Model of Anatomy (Rosse and Mejino 2004; http://fma.biostr.washington.edu), which provides both inheritance (is-a) and merynomic (part-of) trees for elements of human anatomy. Concepts are linked using a mid-sized inventory of properties. In augmenting the OntoSem ontology for use in the medical domain, we have followed the FMA model in certain ways (e.g., with regard to naming conventions) in order to keep our knowledge resources compatible with what we believe will become the accepted standard. However, it would be incorrect to assume that FMA has answered all our needs in the medical domain: whereas it treats only anatomical objects, we need as thorough a treatment of relevant events and their relationship to objects, both anatomical and extra-anatomical.

The current OntoSem ontology contains around 9,000 concepts (objects, events and properties) that are connected through inheritance in a directed acyclic graph and are described by an average of 16 properties each, which can be inherited or locally defined. Most of the concepts belong to the general domain, apart from significant medical subgraphs. The ontology is written in a metalanguage whose atoms resemble English words and phrases (for ease of use by knowledge acquirers) but the semantics of these atoms is distinct from that of the (typically, ambiguous) English words with which they are homographous. When OntoSem is used for text processing applications, ontologically linked lexicons moderate between the text and the ontology. Various versions of the

OntoSem ontology have supported NLP applications in a half dozen languages for which lexicons of up to 50,000 words were compiled.

The inventory of basic primitive ontological properties—attributes and relations—numbers in the hundreds and is growing (though at a slower rate than the overall number of concepts). Domains and ranges are specified for each property. The ontology provides for multivalued selectional restrictions: *sem* for basic semantic constraints, *default* for stronger constraints, *relaxable-to* for the weakest acceptable constraints, and *value* for rigid constraints (*value* is used very rarely in the ontology: it primarily reflects actual properties of concept instances in the VP's long-term memory of assertions, see below).

Within the OntoSem environment, the ontology contains only generalized concepts, not instances of concepts; instances are stored in a fact repository, a model of the agent's long term memory of assertions. This knowledge base uses the same metalanguage as the ontology and is linked to the ontology but is conceptually and formally distinct. Just as the relationships between *types* of objects and events are stored in the ontology (the descriptive component of the knowledge base), the relationships between *instances* of objects and events are stored in the fact repository.

A cornerstone of creating a realistic virtual patient environment is providing for wide variation among instances of patients with a given disease. The basic, ontological model of a disease includes all relevant tracks (i.e., paths of progression), and each track provides many choice points that differentiate cases. Likewise, the basic, ontological model of a human includes all relevant properties of a human with all possible values, from eye color to genetic predisposition to esophageal cancer, to the reaction to all medications and procedures that can be administered through the system. The ontological models of both diseases and humans include default values for all properties, which permits the rapid authoring of patient instances with a focus on the property values that are actually important to the given simulation. For example, in none of the diseases modeled so far does eye color play a diagnostic role; however, every virtual patient instance must have some eye color, so an eye color is automatically attributed to him/her. (Note that if the user *did* ask what a VP's eye color was – once natural language support has been included – the patient could answer, something that might be important for later modeling of diseases like Alzheimer's, where memory tests play a diagnostic role.)

Proto-instances[16] of VPs are data sets for which specific values have been selected (explicitly or automatically) for all human and disease-related properties. A proto-instance is fully prepared to participate in one or more simulations, either by different users or by the same user who seeks to understand how patient outcome will differ with different interventions. A pedagogical utility of proto-instances is that each one can encapsulate a different teaching goal: e.g., "an achalasia patient for whom no treatment options are definitive" or "a GERD patient who will progress to adenocarcinoma if left untreated".

[16] The Knowledge Machine (Clark and Porter, nd) group uses this term with a similar meaning.

An *actual* instance is a proto-instance that is participating, or has participated, in a simulation. For each actual instance, a fact repository is dynamically generated that includes the patient's actual disease state, interventions, etc., over time.

Crucially for the VP project, OntoSem supports the encoding of complex events, also known as scripts, which represent typical sequences of events and their causal and other relationships. Scripts represent how individual events hold well-defined places in routine, typical sequences of events that happen in the world, with a well-specified set of objects that fill the different roles throughout that sequence. For example, if the sequence of events describes a colonoscopy, the participants will include the physician carrying out the procedure, the patient, and any number of other medical personnel; the tools will include the colonosope, various monitors and anesthesia; other props will include the operating table and medical gloves and gowns; events will include anesthetizing the patient, carrying out various procedures with the colonoscope; and so forth. Scripts can contain subscripts (e.g., the scripts of prepping and anesthetizing a patient), and they can be more or less fine-grained, depending on their intended use.

About Scripts

Although many types of knowledge can be represented using the simple slot-filler frames typical of ontologies, applications that rely on high-level reasoning also require scripts, which are descriptions of typical sequences of events, their causal and temporal relationships, and the objects that fill different roles throughout those sequences. Scripts have long been understood as necessary for high-level NLP and AI (e.g., Schank and Abelson 1977) but have been little pursued in practical system building due to the opinion, which we do not share, that the requisite knowledge acquisition is too expensive. Within OntoSem, scripts and simple slot-filler knowledge reside side by side in the ontology. They are employed both in NLP and in simulation/tutoring for essentially the same purpose: to support automatic reasoning.

Scripts represent complex events. There are at least two benefits of including scripts in an ontology: saving time by encoding information once then using across knowledge structures and applications, and giving an automatic reasoner sufficient knowledge upon which to base "intelligent" decisions. Consider the kinds of information that a human would want to be able to assume, rather than have to assert, when writing a script about swallowing: the tongue, larynx, pharynx, esophagus, lower esophageal sphincter and stomach (as well as hundreds of other objects) are all body parts of the same person; all of these are composed of tissue, which is composed of cells, some of which contain a nucleus; tissues are permeated by nerve cells of various kinds; the stomach is distal to the lower esophageal sphincter, which is distal to the body of the esophagus; when a nerve fires it sends a message to the brain… and so on. Once this information is ontologically encoded, a simulator can use it to reason about objects and events in scripts. For example, all variables in all scripts need to be bound; however, the OntoSem simulator can carry out unambiguous variable bindings automatically based on the meronymic relations encoded in the ontology, which turns out to be a significant savings in acquisition time.

We introduce a discussion of issues in developing scripts for simulation using an example. Below is a short excerpt from the script describing swallowing in humans. SWALLOW is the ontological concept that heads the swallowing script: the rest of the script is encoded as the filler of the HAS-EVENT-AS-PART property of SWALLOW as an hierarchical structure describing first the two main subevents of swallowing (OROPHARYNGEAL-PHASE-OF-SWALLOWING and ESOPHAGEAL-PHASE-OF-SWALLOWING), then further expanding those subevents.

The AGENT property of SWALLOW is constrained to HUMAN and the THEME to a BOLUS, which is a small mass of liquid or chewed solid food ready to be swallowed. The PRECONDITION for SWALLOW is that the BOLUS be located in the MOUTH.

The first complex subevent of SWALLOW is the OROPHARYNGEAL-PHASE-OF-SWALLOWING, which moves the BOLUS from the MOUTH to the LARYNX and is caused by a primarily agentive action (by contrast, the ESOPHAGEAL-PHASE-OF-SWALLOWING is unagentive). It has five top-level events: two MOTION-EVENTs, two RELAX-MUSCLE events and one CONTRACT-MUSCLE event (these have subevents as well, like various nerves firing, but we do not detail them here). We assume that the reader can follow the straightforward formalism with the help of the comments presented after semicolons.[17]

```
(SWALLOW                              ;; the head of the script
  (AGENT              HUMAN)
  (THEME       BOLUS)
 (DURATION     10 (DEFAULT-MEASURE  SECOND))
  (PRECONDITION
     (LOCATION
        (DOMAIN          BOLUS)
        (RANGE           MOUTH)))
  (HAS-EVENT-AS-PART
      OROPHARYNGEAL-PHASE-OF-SWALLOWING     ;; expanded below
      ESOPHAGEAL-PHASE-OF-SWALLOWING))      ;; not shown here

(OROPHARYNGEAL-PHASE-OF-SWALLOWING
   (AGENT  HUMAN)
   (THEME  BOLUS)
  (DURATION    1 (DEFAULT-MEASURE  SECOND))
   (HAS-EVENT-AS-PART
     MOTION-EVENT=1            ;; bolus: from mouth to pharynx
     CONTRACT-MUSCLE           ;; contract pharynx, epiglottis closes
     MOTION-EVENT=2            ;; bolus: from pharynx to larynx
     RELAX-MUSCLE=1  ;; cricopharyngeus relaxes
     RELAX-MUSCLE=2))          ;; LES relaxes
```

[17] Our simulation program and NLP processors use exactly this format as input, obviating the need for multiple levels of knowledge representation, one for acquirers and another for machine processing. Currently, knowledge acquisition is a joint effort by a knowledge engineer and subject-matter experts, but we are working on developing a largely machine-initiative knowledge-elicitation system for script writing, relying on the methodologies developed for the Boas KE system (McShane et al. 2003).

```
(MOTION-EVENT=1                ;; bolus: from mouth to pharynx
   (AGENT        HUMAN)
   (THEME          BOLUS)
   (INSTRUMENT           TONGUE)
   (SOURCE       MOUTH)
   (DESTINATION          PHARYNX)
   (DURATION  .3 (DEFAULT-MEASURE  SECOND))
   (EFFECT
      (LOCATION
         (DOMAIN          BOLUS)
         (RANGE           PHARYNX))))

(CONTRACT-MUSCLE               ;; contract pharynx, epiglottis closes
   (AGENT                     HUMAN)
   (THEME                     SET-OF-CONSTRICTOR-MUSCLES-OF-PHARYNX)
   (DURATION             .5 (DEFAULT-MEASURE  SECOND))

   (EFFECT
     (OPENNESS
       (DOMAIN            EPIGLOTTIS)
       (RANGE  0))))

(MOTION-EVENT=2      ;; bolus from pharynx to larynx
   (AGENT                HUMAN)
   (THEME                  BOLUS)
   (INSTRUMENT     SET-OF-CONSTRICTOR-MUSCLES-OF-PHARYNX)
   (SOURCE             PHARYNX)
   (DESTINATION     LARYNX)
   (DURATION  .5 (DEFAULT-MEASURE  SECOND))
   (EFFECT
      (OPENNESS
         (DOMAIN          EPIGLOTTIS)
         (RANGE           1))
      (LOCATION
         (DOMAIN          BOLUS)
         (RANGE           LARYNX))))

(RELAX-MUSCLE =1     ;; cricopharyngeus relaxes
   (AGENT                HUMAN)
   (THEME                CRICOPHARYNGEUS)
   (DURATION  1 (DEFAULT-MEASURE  SECOND))
   (EFFECT
      (PRESSURE
           (DOMAIN       CRICOPHARYNGEUS)
          (RANGE         < 5)))
```

```
(RELAX-MUSCLE=2                ;; LES relaxes
   (THEME        LES[18])
   (DURATION  9 (DEFAULT-MEASURE  SECOND))
   (EFFECT
     (PRESSURE
        (DOMAIN         LES)
        (RANGE          < 5))))
```

Component events in scripts also exist as free-standing concepts in the ontology. However, when used inside a script, such events become much more concrete than the corresponding general concepts: they accept specific fillers of many of their properties instead of the more generic constraints typical for general ontological concepts. For example, the THEME of either of the MOTION-EVENTs in the script excerpt above is restricted to BOLUS, while in the general ontology MOTION-EVENT has a much more generic constraint of PHYSICAL-OBJECT on its THEME property. Thus, components of scripts, though appearing in the ontology, resemble concept instances. Conceptually, they lie somewhere between concepts and actual, real-world instances and are therefore called ontological instances.

Scripts contain ontological instances of both events and objects (e.g., BOLUS in the script is, in fact, an ontological instance of the generic concept BOLUS). Basic ontological concepts provide the necessary information from which the script processor infers variable bindings across the component events of a script (e.g., that both motion events have *the same* BOLUS as their theme).In cases of ambiguity, ontological instances of events and are numbered.

The processor also infers meronymic relationships among ontological object instances. As humans reading the script excerpt above, we probably don't notice that we assume that the TONGUE is located in the MOUTH that is a body part of the HUMAN who is doing the swallowing. Nothing in the script explicitly says that these body parts must be bound to our HUMAN, but in fact they must be interpreted as such if the script is to be understood correctly and support a functional simulation. Requisite anatomical knowledge is recorded in the base ontology and is used by the script processor to infer bindings. For example, when the script processor encounters MOUTH in the SWALLOW script, it finds the shortest ontological path between MOUTH and each of the active ontological instances ("active" instances include any input thematic roles as well as any active instances in the script that called the current script via a HAS-EVENT-AS-PART). In this case, the only two active ontological instances would be the HUMAN and BOLUS instances that are the thematic roles input to the SWALLOW event. For a particular SWALLOW event, perhaps HUMAN-FR228 and BOLUS-22 are involved. The processor will determine that MOUTH has a PART-OF relationship to HUMAN, which is semantically closer than any relationship found between BOLUS and MOUTH. Thus, the processor will infer that the MOUTH in question is the one that is PART-OF HUMAN-FR228.

In cases where bindings can be unambiguously inferred from basic ontological information in this manner, script acquirers need not explicitly indicate them. However, in cases where bindings cannot be unambiguously inferred by the script processor, binding must be overtly indicated. For example, in a later part of the SWALLOW script we need to refer to various STRETCH-RECEPTORs and MOTOR-ENDPLATEs that are located in various body parts. However, there are huge numbers of STRETCH-RECEPTORs and MOTOR-ENDPLATEs throughout the body and, even if we created numerically differentiated ontological instances of them in the script that would not be sufficient to indicate which anatomical structure each was connected to.[19] In such cases, we must explicitly create the bindings in a "bind-variables" field of the given sub-event of the script, as shown below:

```
(PERISTALSIS              ;; moves BOLUS from LARYNX to
  (BIND-VARIABLES                   ;; ESOPHAGEAL-SEGMENT-1
    (STRETCH-RECEPTOR (PART-OF-OBJECT  LARYNX))
    (MOTOR-ENDPLATE=1  (PART-OF-OBJECT  LARYNX))
    (MOTOR-ENDPLATE=2 (PART-OF-OBJECT  ESOPHAGEAL-SEGMENT-1)))
…
```

To reiterate, variable bindings are necessary for many script elements to support the complete and proper interpretation of scripts. However, in those cases when bindings can be unambiguously inferred based on core ontological properties, these bindings need not be overtly written by the script acquirer. Knowing when the script processor will and will not be able to make the correct bindings requires a good understanding of the basic ontological substrate.

The script excerpt above illustrated some basic properties of scripts, like the use of ontological instances, the possibility of embedding sub-events to any depth, the necessity of implicitly or explicitly creating variable bindings, and the anchoring of scripts in the basic ontology with the benefit of being able to rely on properties and values recorded outside of any given script. However, the expressive means described so far represent only a subset of those necessary to support an application like simulation. In this section, we describe a number of others.

Loops

It is frequent in scripts for a given type of event to occur many times with different property values. When such an event is, itself, complex (having subevents as its parts), the desire for economy suggests the need for loops: calls for the same type of complex event with argument values specified for each separate ontological instance. The event of PERISTALSIS, by which muscles of the esophagus (involuntarily) push food through the esophagus to the stomach, is a case in point. According to our conceptual model, the esophagus contains twelve segments whose names correlate with vertebral segments (C6, C7 and T1-T10). The peristalsis events that take the bolus from the larynx to C6, and

[19] We certainly do not want to create a separate concept for each stretch receptor in the human body, differentiated only by which body part they attach to, since such concepts would number in the millions.

from T10 to the stomach, have special properties and are handled separately; however the peristalsis events that take the bolus from C7→T1, T1→T2… T9→10 are essentially the same. We use a special notation to refer to script elements that are part of loops: prefixing them by an asterisk. An excerpt from the ESOPHAGEAL-PHASE-OF-SWALLOWING that shows this convention is as follows:

```
(ESOPHAGEAL-PHASE-OF-SWALLOWING
  (AGENT   *nothing*)          ;; the process is involuntary
  (THEME   BOLUS)
   (DURATION  8 (DEFAULT-MEASURE  SECOND))
  (HAS-EVENT-AS-PART
      …
     *PERISTALSIS
         (SOURCE         C6-SEGMENT-OF-ESOPHAGUS)
         (DESTINATION C7-SEGMENT-OF-ESOPHAGUS)
      *PERISTALSIS
         (SOURCE                    C7-SEGMENT-OF-ESOPHAGUS)
         (DESTINATION T1-SEGMENT-OF-ESOPHAGUS)
    …
     *PERISTALSIS
         (SOURCE                    T9-SEGMENT-OF-ESOPHAGUS)
         (DESTINATION T10-SEGMENT-OF-ESOPHAGUS)
    …  ))
```

When the event *PERISTALSIS is expanded into its subevents, the ontological instances are also referred to using the asterisk notation to show that, during the running of the script, they will be turned into distinct instances of the given concepts:

```
(*peristalsis
  (source        *segment-of-esophagus)
  (destination  *segment-of-esophagus)

  (bind-variables
    (*stretch-receptor
          (part-of-object        *source)
     (*motor-endplate=1
        (part-of-object            *source))
     (*motor-endplate=2
          (part-of-object          *destination))
      …)       ;; more variable bindings

  (has-event-as-part  ;; subevents of *peristalsis
        *stretch
        *fire-nerve
        *stimulate=1
        *stimulate=2
```

```
          *contract-muscle
          *relax-muscle
          *motion-event
          *relax-muscle=2)
   …    )
```

To reiterate, special expressive means are needed for loops to permit economy of knowledge representation while still permitting the script interpreter and simulation program to correctly track variables.

Conditionals

A scripting language must support the use of conditionals, an expressive means that we employed frequently in our modeling of the function and diseases of the esophagus. An example is gastric pressure: each person has a basic gastric pressure, which is within a range of possible values, and this basic pressure is affected by gravity (whether a person is erect or supine) and the amount of food in the person's stomach. If the person is upright, there is no added pressure from gravity, but if he is supine then 5-10 torr is added. If the person has a large amount of food in the stomach (represented as greater than .7 on an abstract scale of quantity) then 7-10 torr is added, and if he has less food in the stomach then less pressure is added. This information is recorded in scripts as follows:

```
(stomach
   (pressure (sum        (pressure-basic-gastric
                          pressure-from-gravity
                          pressure-from-food)))[20]
   (pressure-basic-gastric (<> 5 10))
   (pressure-from-gravity
     (if
       (spatial-orientation
          (domain        human)
          (range vertical))
       then  0)
     (if
        (spatial-orientation
           (domain       human)
           (range        horizontal))
       then  (<> 5 10))))
   (pressure-from-food
     (if
        (location
          (domain        stomach)
          (range         (food (quant  (> .7)))))
       then  (<> 7 10))
    (if
```

[20] Abdominal pressure is another variable but we omit it in this exposition.

```
        (location
           (domain        stomach)
           (range (food   (quant  (<>  .3 .6)))))
     then  (<> 4  6.9))
    (if
        (location
           (domain        stomach)
           (range (food (quant  (< .3)))))
     then  (< 4)))))
```

Within larger scripts of esophageal function, the various values for gastric pressure are combined with values for the pressure of the LES (lower esophageal sphincter) to determine whether or not a person will experience reflux. If he does, then he will experience pain, with the duration of the reflux affecting the severity of pain. If the reflux continues over a long period of time, the esophagus will enter a disease state in which its actual properties change permanently. If the disease is treated (through medications) at an early stage, the esophagus will heal to an extent but if treated at a later stage, healing will be less complete or impossible; and so on. All of these conditionals, and many more, are included in our current swallowing script, which has been implemented as a simulation.

Time

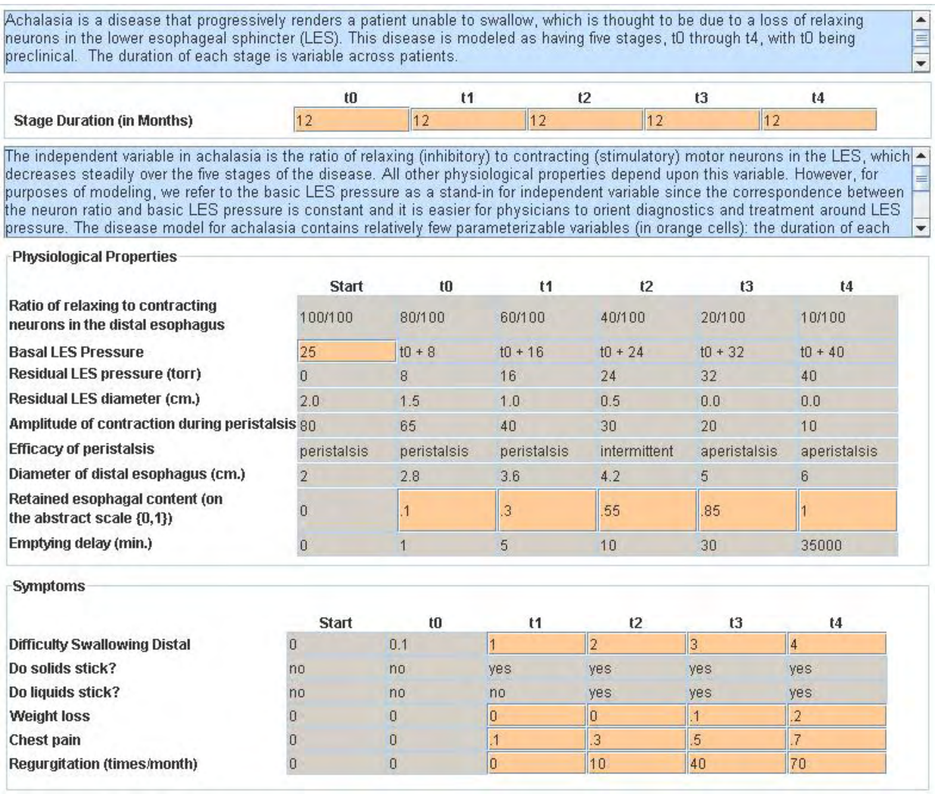
Time is an essential element of scripts, especially if they are used for simulation. In the case of swallowing, the entire process takes around 10 seconds, with different subscripts accounting for different portions of that, as indicated by the DURATION slot.

In medical simulation, modeling time is necessary in order to account for the progression of disease. An example is the growth of a tumor. Once a tumor is assigned a growth rate, it will grow at that rate when the script is run. Physiological effects of the tumor at different sizes are anticipated in the script. For example, a small tumor in the esophagus will show no symptoms, a medium-sized tumor will lead to mild dysphagia (difficulty swallowing), a large tumor will block solids but still permit liquids to go down, and a massive tumor will block all substances from passing. The notions of small, medium and large are determined by the physicians acting as subject-matter experts during script development.

Patient authoring

In addition to acquiring general medical knowledge, the project also requires the creation of an extensive library of actual virtual patients that exhibit a variety of diseases and genetic and cognitive traits that influence disease progression and response to treatment. We have developed an advanced interactive environment to support patient authoring by physicians who may not know about the internal workings of our models of disease and treatment.

The patient creation process for all diseases begins with providing basic information about the patient: name, age, gender, weight, race, etc. We omit this aspect of the interfaces, as well as other aspects that are easily described in prose, in the screen shots below for reasons of space.



Achalasia is a disease that progressively renders a patient unable to swallow, which is thought to be due to a loss of relaxing neurons in the lower esophageal sphincter (LES). This disease is modeled as having five stages, t0 through t4, with t0 being preclinical. The duration of each stage is variable across patients.

	t0	t1	t2	t3	t4
Stage Duration (in Months)	12	12	12	12	12

The independent variable in achalasia is the ratio of relaxing (inhibitory) to contracting (stimulatory) motor neurons in the LES, which decreases steadily over the five stages of the disease. All other physiological properties depend upon this variable. However, for purposes of modeling, we refer to the basic LES pressure as a stand-in for independent variable since the correspondence between the neuron ratio and basic LES pressure is constant and it is easier for physicians to orient diagnostics and treatment around LES pressure. The disease model for achalasia contains relatively few parameterizable variables (in orange cells): the duration of each

Physiological Properties

	Start	t0	t1	t2	t3	t4
Ratio of relaxing to contracting neurons in the distal esophagus	100/100	80/100	60/100	40/100	20/100	10/100
Basal LES Pressure	25	t0 + 8	t0 + 16	t0 + 24	t0 + 32	t0 + 40
Residual LES pressure (torr)	0	8	16	24	32	40
Residual LES diameter (cm.)	2.0	1.5	1.0	0.5	0.0	0.0
Amplitude of contraction during peristalsis	80	65	40	30	20	10
Efficacy of peristalsis	peristalsis	peristalsis	peristalsis	intermittent	aperistalsis	aperistalsis
Diameter of distal esophagus (cm.)	2	2.8	3.6	4.2	5	6
Retained esophagal content (on the abstract scale {0,1})	0	.1	.3	.55	.85	1
Emptying delay (min.)	0	1	5	10	30	35000

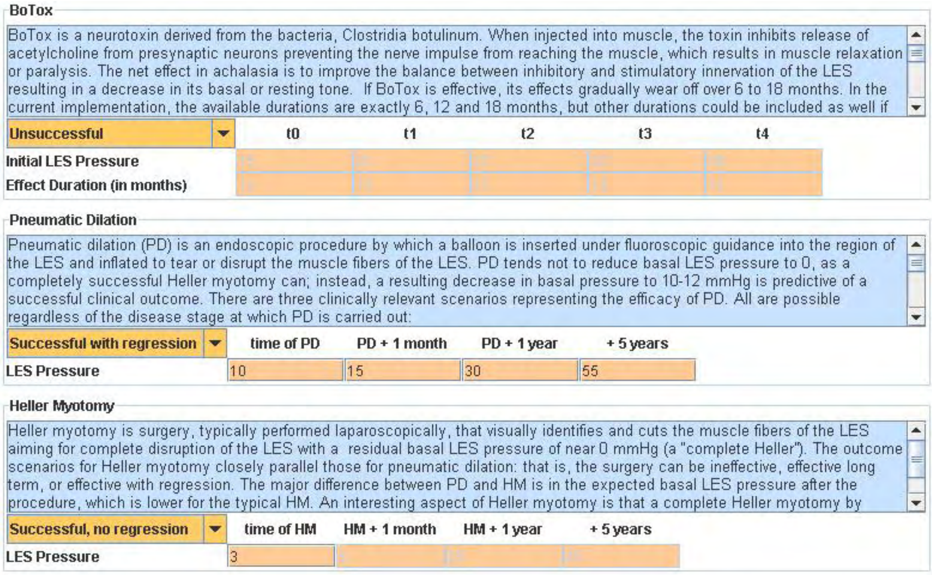
Symptoms

	Start	t0	t1	t2	t3	t4
Difficulty Swallowing Distal	0	0.1	1	2	3	4
Do solids stick?	no	no	yes	yes	yes	yes
Do liquids stick?	no	no	no	yes	yes	yes
Weight loss	0	0	0	0	.1	.2
Chest pain	0	0	.1	.3	.5	.7
Regurgitation (times/month)	0	0	0	10	40	70

Figure 13. An excerpt from the authoring interface for achalasia.

Disease models break down into two major classes based on whether or not the physiological causal chains underlying the disease are well understood. In cases where physiological causal chains are relatively poorly understood – as for achalasia, scleroderma esophagus and Zenker's diverticulum – the simulation is primarily driven by temporal causal chains. Each disease is divided into conceptual stages, with each stage being associated with clinically observed physiological changes and symptom profiles. As simulated time passes, the patient's state changes incrementally, calculated using an interpolation function that incorporates the start value of each property at the beginning of the disease and the end value for each conceptual stage. Figure 13 shows these aspects of the model of achalasia, as presented to patient authors. The text in the blue background explains each aspect of the model using methods of progressive disclosure (a small text field with a scroll bar), which permits users with different levels of experience using the system to use the same interface without the explanatory materials becoming cumbersome. The explanatory text conveys important aspects of how the recorded property values are interpreted within the model and processed by the simulation engine, like which property values impaired by a disease can be reversed given an effective treatment, and which variables are independent and which are dependent. In short, the knowledge used by the simulator goes well beyond what is needed to parameterize a new patient instance, and it is conveyed to patient authors as text in order to clarify – albeit in encapsulated form – how the instantiated model works.

The gray cells indicate values that are fixed for all patients, since permitting their variation is not necessary for either of our immediate goals: (a) generating automatic function in the simulation (e.g., if a given biological pathway can be affected by medication, then it must be parameterizable) and (b) permitting noteworthy variation among patients within a teaching context. The orange cells indicate property values that can be changed for each patient, within ranges visible by mousing over the given cell. This division between paramaterizable and non-parameterizable property values points up an important benefit of making our models accessible to the community: for the current teaching application, it was appropriate to make certain property values parameterizable within certain ranges; however, for some other application it might be necessary to make more of these values more variable across patients, which can be readily done with no changes required of the simulation engine.



BoTox

BoTox is a neurotoxin derived from the bacteria, Clostridia botulinum. When injected into muscle, the toxin inhibits release of acetylcholine from presynaptic neurons preventing the nerve impulse from reaching the muscle, which results in muscle relaxation or paralysis. The net effect in achalasia is to improve the balance between inhibitory and stimulatory innervation of the LES resulting in a decrease in its basal or resting tone. If BoTox is effective, its effects gradually wear off over 6 to 18 months. In the current implementation, the available durations are exactly 6, 12 and 18 months, but other durations could be included as well if

Unsuccessful	t0	t1	t2	t3	t4
Initial LES Pressure					
Effect Duration (in months)					

Pneumatic Dilation

Pneumatic dilation (PD) is an endoscopic procedure by which a balloon is inserted under fluoroscopic guidance into the region of the LES and inflated to tear or disrupt the muscle fibers of the LES. PD tends not to reduce basal LES pressure to 0, as a completely successful Heller myotomy can; instead, a resulting decrease in basal pressure to 10-12 mmHg is predictive of a successful clinical outcome. There are three clinically relevant scenarios representing the efficacy of PD. All are possible regardless of the disease stage at which PD is carried out:

Successful with regression	time of PD	PD + 1 month	PD + 1 year	+ 5 years
LES Pressure	10	15	30	55

Heller Myotomy

Heller myotomy is surgery, typically performed laparoscopically, that visually identifies and cuts the muscle fibers of the LES aiming for complete disruption of the LES with a residual basal LES pressure of near 0 mmHg (a "complete Heller"). The outcome scenarios for Heller myotomy closely parallel those for pneumatic dilation: that is, the surgery can be ineffective, effective long term, or effective with regression. The major difference between PD and HM is in the expected basal LES pressure after the procedure, which is lower for the typical HM. An interesting aspect of Heller myotomy is that a complete Heller myotomy by

Successful, no regression	time of HM	HM + 1 month	HM + 1 year	+ 5 years
LES Pressure	3			

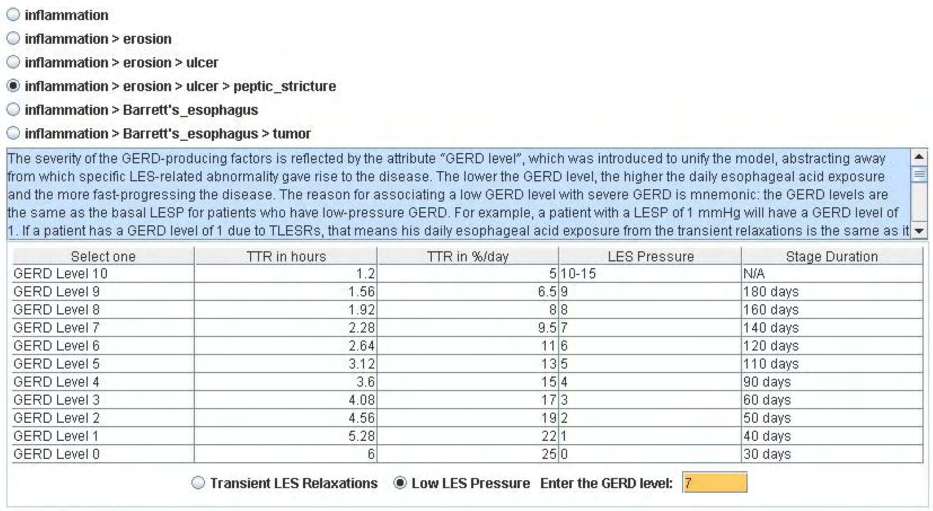
Figure 14. Treatment outcomes for achalasia patients.

The remaining aspect of patient parameterization for achalasia regards treatments. There are three treatment options, each explained in the associated blue shaded text fields (see Figure 14). Each has three potential options: unsuccessful, successful with regression, and successful without regression. If a treatment is unsuccessful, as for BoTox in Figure 14, there are no further choices to be made: the patient's condition is unchanged (note that the values in the cells of the corresponding table are grayed out). If a treatment is successful with regression (as for pneumatic dilation in Figure 14), the author must choose the rate of regression of the basal pressure of the lower esophageal sphincter (LESP) over time: its value one month, one year and five years after the procedure. (Although LESP is actually dependent upon the ratio of contracting to relaxing neurons, it is conceptually easier for clinicians to reason using LESP). After a procedure, most physiological properties and symptoms retain the original correspondences with LESP shown in the tables in Figure 13; however, the efficacy of peristalsis and diameter of the distal esophagus never improve once compromised, as explained in the blue text field. If a treatment is successful without regression (as for Heller myotomy in Figure 14), only the original post-procedure LESP must be indicated, with most other property values following suit, as described above.

The other class of diseases modeled in the system are those for which physiological causal chains are quite well understood. GERD, LERD and LERD-GERD are all of this type. For reasons of space, we highlight just one aspect of the causal modeling of GERD and how it is reflected in the patient authoring interface (see McShane et al. 2007a,b for more in-depth descriptions of these disease models).

GERD can be defined as any symptomatic clinical condition that results from the reflux of stomach or duodenal contents into the esophagus. Based on a person's inherent predispositions (no biomarkers have yet been discovered), the disease can take one of six paths, shown at the top of Figure 15. The author selects one path for his patient, which sets associated property values in the patient. The two sources of GERD are abnormally low pressure of the lower esophageal sphincter (LES) (< 10 mmHg), or an abnormally large number or duration of transient relaxations of the LES (TLESRs), both of which result in increased acid exposure of the esophageal lining. The text in blue in Figure 15 (which is quite long; note the slider size) describes how LESP and/or TLESRs are used as independent variables in the model. We repeat an excerpt from that text here as an example of how text complements the formal aspects of disease models:

> The severity of the GERD-producing factors is reflected by the attribute "GERD level", which was introduced to unify the model, abstracting away from which specific LES-related abnormality gave rise to the disease. The lower the GERD level, the higher the daily esophageal acid exposure and the more fast-progressing the disease. The reason for associating a low GERD level with severe GERD is mnemonic: the GERD levels are the same as the basal LESP for patients who have low-pressure GERD. For example, a patient with a LESP of 1 mmHg will have a GERD level of 1. If a patient has a GERD level of 1 due to TLESRs, that means his daily esophageal acid exposure from the transient relaxations is the same as it would have been if he had had a basal LESP of 1. Using GERD level as the anchor for modeling provides a simple mechanism for incorporating a patient's lifestyle habits into the simulation: whenever he is engaging in bad lifestyle habits (assuming he has GERD-related sensitivities to those habits), his GERD level decreases by 1. For patients with a baseline GERD level of 10 – which is not a disease state – this means that engaging in bad habits is sufficient to initiate GERD and discontinuing them is sufficient to promote healing without the need for medication. For patients with a baseline GERD level of less than 10, lifestyle improvements can slow disease progression but not achieve the healing of previous esophageal damage.



○ inflammation

○ inflammation > erosion

○ inflammation > erosion > ulcer

● inflammation > erosion > ulcer > peptic_stricture

○ inflammation > Barrett's_esophagus

○ inflammation > Barrett's_esophagus > tumor

The severity of the GERD-producing factors is reflected by the attribute "GERD level", which was introduced to unify the model, abstracting away from which specific LES-related abnormality gave rise to the disease. The lower the GERD level, the higher the daily esophageal acid exposure and the more fast-progressing the disease. The reason for associating a low GERD level with severe GERD is mnemonic: the GERD levels are the same as the basal LESP for patients who have low-pressure GERD. For example, a patient with a LESP of 1 mmHg will have a GERD level of 1. If a patient has a GERD level of 1 due to TLESRs, that means his daily esophageal acid exposure from the transient relaxations is the same as it

Select one	TTR in hours	TTR in %/day	LES Pressure	Stage Duration
GERD Level 10	1.2	5	10-15	N/A
GERD Level 9	1.56	6.5	9	180 days
GERD Level 8	1.92	8	8	160 days
GERD Level 7	2.28	9.5	7	140 days
GERD Level 6	2.64	11	6	120 days
GERD Level 5	3.12	13	5	110 days
GERD Level 4	3.6	15	4	90 days
GERD Level 3	4.08	17	3	60 days
GERD Level 2	4.56	19	2	50 days
GERD Level 1	5.28	22	1	40 days
GERD Level 0	6	25	0	30 days

○ Transient LES Relaxations ● Low LES Pressure Enter the GERD level: 7

Figure 15. An excerpt from the patient authoring process for GERD.

As is clear by the table, when the author selects the GERD Level (he chose 7 in Figure 15) the duration of each stage of the disease and the total time in reflux (TTR) are automatically selected for him. Other aspects of the patient authoring interface permit authors to select lifestyle habits for their patients, whether those lifestyle habits affect their GERD, their symptom profile and their response to medications. Our point in this example is to show that even when a disease is modeled using causal chains that are encoded in quite complex ontological scripts and realized in even more complex simulation programs, the conceptual substrate of the basic models can readily be shared with – and contributed to – by the larger community.

Discussion

The patient authoring interface in MVP highlights key aspects of the cognitive model of each disease, providing patient authors with explanations of the choice space without either repeating all the information about each disease available in textbooks or expounding upon the implementation of the simulation engine. The core aspects of each disease model include which property values are parameterizable among patients and which ones are fixed for all patients, what ranges of values are permitted for each property at each stage of the disease (seen by rolling over cells in the interface), how "healing" is interpreted with respect to each property value affected by a disease, and how parameterizable property values are used to "bridge" unknown aspects of diseases, like as yet undiscovered genetic influences. The grain-size of description – including which aspects are made parameterizable and which physiological causal chains are included in the model – is influenced by the given application but could easily be changed to suit other applications using our ontologically grounded knowledge encoding methodology. Let us consider this last point in more detail using the example of GERD. As shown in Figure 15, by selecting a GERD level, the author automatically sets the duration of each conceptual stage of the disease and the total time in reflux per day. For a pedagogical application, these fixed correspondences are very useful. However, we plan to use this knowledge environment for other applications as well, like automatically analyzing electronic patient records both to validate the model and to learn new population-level clinical knowledge. It is likely that some patients fall outside the range of expected outcomes of our current model, which can lead in two directions: either expanding the current model by making more aspects parameterizable, or creating a second, non-pedagogical version of GERD, thus permitting the pedagogical version to retain strong correspondences that are useful as a conceptual architecture, abstracting away from confounding cases. In fact, our knowledge environment can accommodate any number of versions of a disease model suited to different applications. Similarly, the mentoring module (which we did not discuss here) can also accommodate any number of versions: currently, our virtual mentor reflects one set of clinical preferences, but there could be a entire population of virtual mentors reflecting variations on clinical management practices. Recording disease and mentoring models using ontological scripts (rather than, for example, very large rule sets) permits such variation to be readily recorded and managed.

We have designed our knowledge environment such that we can readily collaborate with the broader community. For example, as more genetic influences on diseases are discovered, and more causal chains are understood, these will be used to replace temporal causal chains with physiological ones. We have made our disease models inspectable so that not only can experts assess them in terms of how virtual patients behave in a simulation, but also in terms of the core tenets of the mental models of the physicians who contributed to their development.

Language Processing

The OntoSem environment that provides the knowledge substrate for the MVP project has been first developed to support natural language processing. So, it is not surprising that OntoSem supports NLP in the MVP project, too. In this section, we will very briefly describe the NLP tasks within the MVP project.

Figure 16 shows the architecture of the OntoSem text understanding system (e.g., Nirenburg and Raskin 2004; Beale et al. 1995, 2003, 2004; McShane et al. 2005a, 2006). Text analysis involves preprocessing, syntactic analysis, semantic analysis, and discourse processing (including contextual compositional semantics for, among other things, reference resolution). The static resources used by the system include: (1) A property-rich, general-purpose ontology of about 9,000 concepts that contains information about the types of entities and events known to the system, together with their properties; (2) An ontologically-linked lexicon of about 20,000 words and phrases that includes syntactic and semantic information to support disambiguation; (3) An onomasticon, or lexicon of proper names; and (4) A repository of the given language processing agent's long-term memory of assertions (LTM-A). Although ours is a general-purpose ontology, coverage of the medical domain has been expanded for the MVP application to include detailed descriptions of human anatomy, complex events (i.e., scripts) reflecting normal and pathological physiological processes (cf. Schank and Abelson 1977), and best clinical practices for the diagnosis and treatment of diseases.

The primary goal of OntoSem analysis is to automatically generate unambiguous text-meaning representations (TMRs) of input, represented in a metalanguage that is grounded in the OntoSem ontology. Semantic analysis begins with word sense disambiguation and the establishment of basic semantic dependencies. This process relies on knowledge about selectional restrictions – mutual constraints – that is stored in the lexicon and ontology entries that are activated by the input text. In OntoSem, selectional restrictions are multi-valued, which allows for contextual tightening or relaxation of constraints when building TMRs. When successful, this process yields *basic* TMRs. However, no lexicon or ontology, no matter how broad and deep, can guarantee successful disambiguation of all inputs. This is why OntoSem incorporates a number of methods and algorithms for dealing with cases of residual ambiguity and zero output. These include a) an algorithm for the dynamic tightening and relaxing of selectional restrictions based on the properties of the input text; b) a statistically trained disambiguation algorithm based on comparing weighted distances between pairs of concepts in the ontology; c) an algorithm for the unidirectional application of selectional restrictions to process input words that are not listed in the system's lexicon; and d) an algorithm for deriving TMRs from fragments of input in cases of failure to produce a TMR for a complete sentence. The use of both knowledge-based and empirical methods demonstrates that OntoSem takes a practical, task-oriented approach to text analysis rather than a method-oriented one.

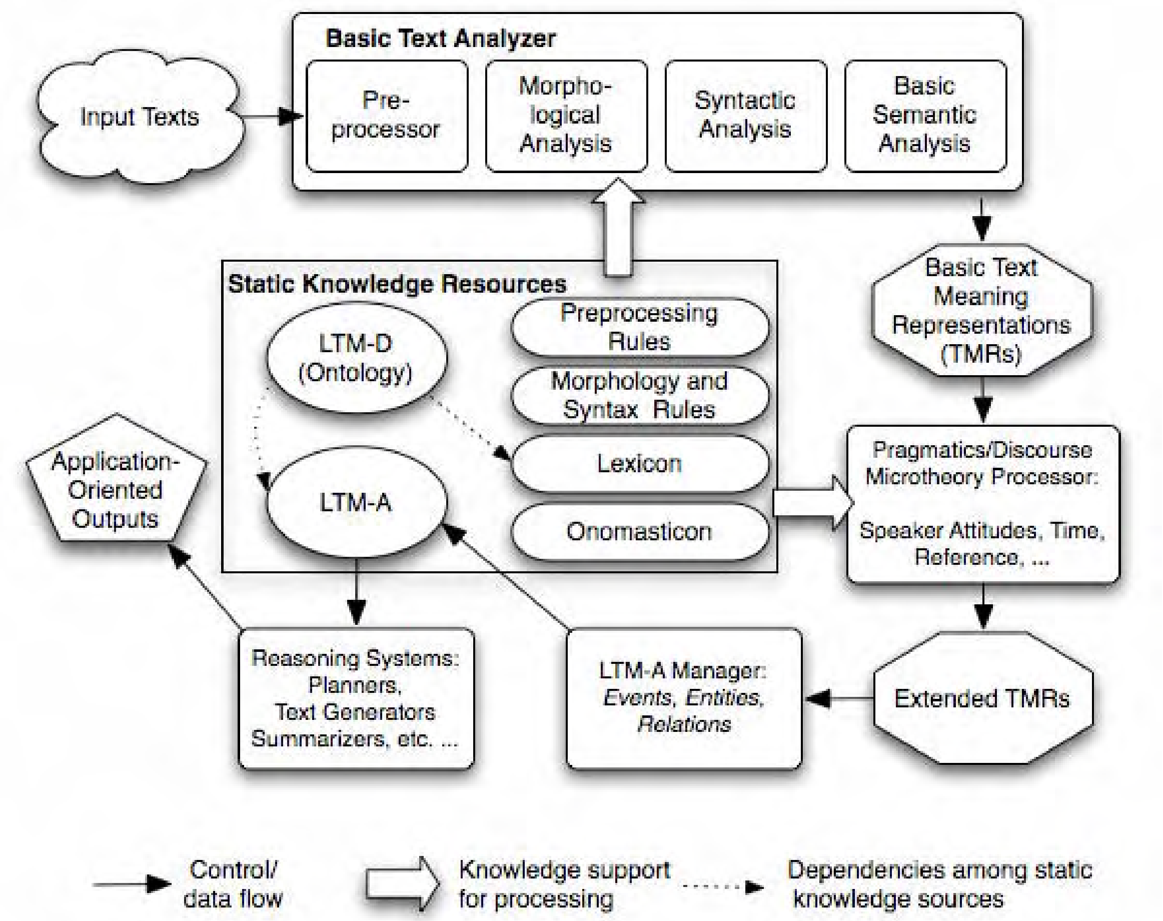

Basic Text Analyzer
Input Texts
Pre-processor
Morpho-logical Analysis
Syntactic Analysis
Basic Semantic Analysis
Static Knowledge Resources
LTM-D (Ontology)
Preprocessing Rules
Morphology and Syntax Rules
Lexicon
LTM-A
Onomasticon
Basic Text Meaning Representations (TMRs)
Pragmatics/Discourse Microtheory Processor:
Speaker Attitudes, Time, Reference, ...
Application-Oriented Outputs
Reasoning Systems: Planners, Text Generators Summarizers, etc. ...
LTM-A Manager: Events, Entities, Relations
Extended TMRs
Control/ data flow
Knowledge support for processing
Dependencies among static knowledge sources

Figure 16. The architecture of the OntoSem text analyzer. Note the use of the long-term memory of descriptions (the ontology) and the long-term memory of assertions.

Once a basic TMR is produced, the analyzer attempts to generate an *extended* TMR by carrying out advanced aspects of text analysis, formulated as "microtheories." In the proposed project we seek to enhance the microtheories of reference resolution and unexpected input processing. Among other benefits, this will enhance the agent's ability to leverage information in text for the population of its LTM-A.

All development and evaluation in OntoSem is supported by DEKADE, our Development, Evaluation, Knowledge Acquisition and Demonstration Environment (McShane et al. 2005c). Among DEKADE's many functions are: accepting text input to be run through the analyzer; permitting the results of each stage of analysis (preprocessing, syntax, semantics, procedural semantics) to be viewed and edited using textual or graphical interfaces; offering easy access to the knowledge resources (lexicon, ontology, onomasticon and LTM-A) so that they can be edited and supplemented during work on a text or using other acquisition methodologies; and supporting formal evaluation of the resulting TMRs (see Nirenburg et al. 2004 for results of our first evaluation). The DEKADE environment, completed only recently, has drastically increased the speed of OntoSem system development and the efficiency of system evaluation.

OntoSem incorporates a number of earlier processors – the NMSU CRL TIPSTER tokenizer (Davis and Ogden 2000), the BBN IdentiFinder tokenizer (Bikel et al. 1999), the Bikel parser (Bikel 2002), the Hunter-Gatherer constraint satisfaction architecture (Beale 1997) and the OntoSearch algorithm for finding cheapest paths between two concepts in the OntoSem ontology (Onyshkevych 1997). A number of existing knowledge resources were used by OntoSem knowledge engineers to help them acquire the ontology, the lexicon and other knowledge resources. These include the XTAG grammar (www.cis.upenn.edu/~xtag/), the Penn Treebank (www.cis.upenn.edu/~treebank/), the Foundational Model of Anatomy (sig.biostr.washington.edu/projects/fm/) and WordNet (wordnet.princeton.edu).

NLP Dialog in MVP

The dialog-enhanced MVP system, MVP-D, is under development at the moment of writing. It will supplant the menu-based interaction between the user and the cognitive agents in the system – the virtual patient and the tutor. In MVP-D (as opposed to MVP), the VP will not have direct access to its physiological state. Instead, like a real person, it will perceive its own "body", including sensations of pain and other symptoms, as warranted. The VP will thus be able to reason about whether to consult a physician, whether to stick to a medication regimen, etc. The VP will also remember the gist of its prior conversations with the user (the attending physician) as well as the beliefs/facts about its physiology that it accumulates over time in its LTM-A. The communication with the user (the attending physician) will be conducted in natural language. Natural language will also be used for the communication between the tutor and the user. The tutor will have access to (i.e., memory of) all of the past dialog interactions in the system, the current state of the student's knowledge about the patient, and the knowledge base of best practices. The first version of MVP-D will operate using the current repertoire of the 6 esophageal diseases supported by MVP. In parallel to the proposed work, and under different funding, we will continue work on acquiring knowledge about heart disease for the VP. Once such knowledge is acquired, MVP-D will be evaluated on an extended knowledge base and broadened dialog coverage.

Learning new concepts

A core component of MVP is the module supporting dialog between the VP and the student. One of the important aspects of creating natural-sounding dialog is to allow different VPs to have different ways of expressing themselves, including different degrees of knowledge about medicine and medical terminology. Students should be as adept at conducting an interview with an uninformed patient (in which case many paraphrases of medical terms might be needed) as with a fellow physician. Another important aspect of conducting an interview is educating the VP about his condition, his medication regime, needed lifestyle improvements, options for treatments, and so on. The result of all of this communication has to be learning on the part of the VP: after all, if the student chooses to explain to the VP that the feeling of having something stuck in one's throat is called a "globus sensation", we would expect the VP to remember that 2 minutes later (and, perhaps, at the next visit as well). At a minimum, the patient will eventually need to remember the name of his disease, his medications, and so on.

In developing the NLP support for MVP, dynamically adding to the VPs knowledge repository is of primary concern, and is a good example of an adaptive system module. That is, not only will the patient learn about its condition and treatment through verbal interactions, it will be able to put this knowledge to use in decision-making, one of the key functionalities of the cognitive agent. For reasons of space, in this paper we use a sample interaction to give an informal, content-oriented description of this adaptive process – a more formal analysis would have required a description of the various static and dynamic knowledge resources underlying the system and, specifically, its natural language processing component (this information can be found in [6], among others).

Suppose that during a patient interview the student asks the VP if the latter ever experiences regurgitation, and suppose that the VP does not understand the term *regurgitation* (that is, the entry for this word is absent from the VP's semantic lexicon).

Background: Each agent in the system is supplied with its own version of the knowledge resources available to the system. This means, for example, that the tutor knows much more about diseases and clinical practices than the virtual patient. The latter's lexicon and ontology are deliberately filtered to reflect an average lay person's knowledge of medicine. During patient authoring, the author selects the level of medical knowledge of the patient, and the lexicon and ontology supplied to the patient are populated accordingly.

The VP will ask for clarification by issuing a dialog turn such as: "What is regurgitation?" The goal of this subdialog is to learn the new term.

First let us consider the eventuality when the human user responds by suggesting a synonym for regurgitation. If the synonym is in the patient's lexicon, then the VP first learns the new lexicon entry, whose semantics will be the same as for the known synonym, and then responds to the original question using this semantic interpretation. As a result of this process, the VP's lexical stock is increased, so that the next time the formerly unknown word is used in a dialog, there will be no need for the clarification subdialog. Of course, the student might provide a synonym that is not in the patient's lexicon, in which case the patient may opt to continue the clarification subdialog.

Instead of a synonym, the user may provide a description of what regurgitation is. On receiving this input, the patient's goal is to match the description to a concept in its ontology. If such a concept is found, then a new lexicon entry for the unknown word (in this case, regurgitation) is created, and the matching concept is used in the entry's semantic description. If no concept is a close enough match, then the VP must learn a new concept.

Suppose the student supplies the following definition of regurgitation: "The return of partially digested food from the stomach to the mouth."[21] And suppose the VP knows all the words in the above explanation.[22] The language analyzer processes this input and comes up with the following text meaning representation (only relevant information is presented; ontological concepts are shown in small caps):

RETURN-1 (which IS-A MOTION-EVENT)

THEME	INGESTIBLE-105
SOURCE	STOMACH-1
DESTINATION	MOUTH-1

The above text meaning representation contains numbered instances of ontological concepts as heads of frames and values of properties. But when comparing this structure with the concepts in the patient's ontology, we disregard instance numbers, in effect treating this text meaning representation as a candidate ontological concept. If a sufficiently close match is found in the VP's current ontology, then the putative new ontological concept is discarded and the already existing best match is used to describe the semantics of the unknown word. If the best match is not considered close enough, then the candidate concept is "promoted" to a regular concept in the ontology. In our example, the search in VP's "lay person" ontology for the best match of the above text meaning representation is the concept VOMIT:[23]

	VOMIT
IS-A	ANIMAL-SYMPTOM, MOTION-EVENT
THEME	INGESTIBLE
SOURCE	STOMACH
DESTINATION	MOUTH
VELOCITY	> .8

[21] This definition is a real one from The American Heritage Science Dictionary.

[22] This is actually the case with our language processing resources – the complete (unfiltered) English lexicon in our system covers over 30,000 word senses, and the ontology used to explain these senses consists of over 9,000 concepts, each of which has on average 16 properties defined for it.

[23] The process briefly described here is a special case of learning ontologies and lexicons by reading text and analyzing it using our OntoSem environment for meaning extraction. This work is described in more detail in [10].

In other words, the event most closely associated with food coming up that the patient knows about is vomiting. If the two concepts are judged sufficiently similar, the concept VOMIT will be used to describe the semantics of regurgitation. If not, the candidate concept will be included in the ontology and given the name REGURGITATION. If the results of clustering are uncertain, the VP may opt for a continuation of the clarification subdialog by passing the responsibility for this decision to the user, that is, by asking, e.g., "Do you mean vomiting?" If the user agrees that these are sufficiently close, that settles the issue. But if the user considers it important to distinguish between vomiting and regurgitating, then he or she will respond to the effect that regurgitation is like vomiting but not as forceful. In this case, the learning module in the VP will add the concept REGURGITATION to the ontology; this new concept will be similar to VOMIT but have the additional property that its VELOCITY is lower (the property VELOCITY will be licensed for VOMIT on account of its being an ontological descendant of MOTION-EVENT, for which the property of VELOCITY is defined).

It is important to stress that the "maximum coverage" ontology that our system can use already has the concept REGURGITATION, along with its siblings VOMIT and REFLUX. All of the above concepts are, in fact, children of BACKWARDS-MOTION-OF-INGESTED-SUBSTANCE, which itself is a child of both ANIMAL-SYMPTOM and MOTION-EVENT (exploiting multiple inheritance). The experiencer of all of these events is a MEDICAL-PATIENT, but the events differ with respect to five properties, as shown in the following table illustrating a subset of the knowledge in the "maximum-coverage ontology."

	REGURGITATE	**VOMIT**	**REFLUX**
IS-A	BACKWARDS-MOTION-OF-INGESTED-SUBSTANCE		
THEME	BOLUS	CHYME	CHYME
SOURCE	ESOPHAGUS	STOMACH	STOMACH
DESTINATION	THROAT	MOUTH	ESOPHAGUS
VELOCITY	< .2	> .8	< .2
INSTRUMENT	-	MUSCLE-LAYER	-

The above underscores our commitment to adaptivity in the system. Indeed, we could have made the VP omniscient, at least in the domain of the diseases that it carries. Instead, we chose to model a much more realistic situation without "cheating" by allowing all agents to operate with full access to all knowledge at all times.

Utility Beyond the Current Application

The MVP project can be viewed as just one of a number of applications in the area of intelligent clinical systems. The latter, in turn, can be viewed as one of the possible domains in which one can apply modeling teams of intelligent agents featuring a combination of physical system simulation and cognitive processing.

So, in the most general terms, our work can be viewed as devoted to creating working models of societies of artificial intelligent agents that share a simulated "world" of an application domain with humans in order to jointly perform cognitive tasks that have until now been performed exclusively by humans. Sample applications of such models include:

- a team of medical professionals diagnosing and treating a patient (with humans playing the role of either a physician or a patient)
- a team of intelligence or business analysts collecting information, reasoning about it and generating analyses or recommendations (with humans playing the role of team leader)
- a team of engineers designing or operating a physical plant (with humans playing the role of team leader)
- a learning environment (where humans play the role of students).

As can be seen, this work is at the confluence of several lines of research – cognitive modeling, ontological engineering, reasoning systems, multi-agent systems, simulation and natural language processing.

A basic development and delivery environment for this work should consist of at least:

1. the working model of a society of intelligent agents;
2. a simulated physical system; and
3. an interface for communicating with the human members of the society.

The intelligent agents must be endowed with knowledge about

a) their world;
b) their own character traits;
c) a set of goal types;
d) a set of plans to achieve each of the goals; and
e) their experiences, in the form of memories of past actions and states of
    i) the physical system;
    ii) other agents in the environment; and
    iii) themselves.

In a typical application, these models must be fully or partially shared among the agents in the society. The agents are capable of

1. (simulated) perception of changes in the (simulated) physical system;
2. understanding communications from other agents (as part of natural language dialog)
3. reasoning about the state of the physical system, themselves and other agents;
4. deciding on what to do next on the basis of the state of the world and their own goals and plans;
5. performing (simulated) physical actions in the world;
6. performing communicative actions (as part of natural language dialog).

Natural language processing (NLP) is important in such an undertaking because, unlike the majority of multi-agent systems currently under development, our work always presupposes the participation of people, not only artificial intelligent agents. Natural language is the most natural way of communicating for people, even if having to use it introduces an additional complexity to building agent systems. Our approach to NLP is knowledge-based and is fully compatible with our approach to general cognitive modeling, reasoning and problem solving. The meanings of natural language words and expressions are encoded in our approach using the same ontology that describes the general world model of the agents. The model of dialog we use adopts the same goal- and plan-oriented apparatus as our general problem solving architecture. This brings about economies of scale in developing the overall environment.

Constraining the above work to the medical domain results in a narrowing of scope that enhances feasibility. The types of artificial intelligent agents in a medicine-oriented environment include attending physicians, consulting physicians, lab technicians and patients. The world model in this application consists of three components:

1. the model of the human body and its associated physiological and pathological processes (the physical system);
2. the model of clinical practices for each type of medical professional involved (this model specifies the typical professional goals and plans of medical professionals);
3. the cognitive model of the patient, to be coupled with the model of the physical system.

Applications of the medicine-oriented environment may include:

1. Assisting medical professionals in finding needed references, answers to questions, etc. about diseases, their diagnostics and treatment from online sources.
2. Compiling a database of semantically analyzed patient instances drawn from available databases of patient records. These patient instances can be used as training materials, reference materials, or to validate the models being developed in the environment.
3. Compiling a database of clinical practices that can be queried by healthcare professionals.
4. Creating patient information materials, including simulations of what their disease might look like if they continue their current lifestyle, information on how to improve their health given a certain disease, and other patient maintenance-related issues.
5. Providing automatic interactive environments for the use of medical examination boards.

MVP is the first application of the above environment that we have undertaken.

Discussion

How does our modeling differ from other types of modeling?

Computer models can be classified according to several parameters, including whether they are:

A. Stochastic or deterministic

Deterministic Models take no account of random variation and therefore give a fixed and precisely reproducible result. They can be implemented by numerical analysis or computer simulation. Deterministic models are often described by sets of differential equations.

Stochastic Models are mathematical models, which take into consideration the presence of some randomness in one or more of its parameters or variables. The predictions of the model therefore do not give a single point estimate but a probability distribution of possible estimates.

The essential difference between a stochastic and deterministic model is that in a stochastic model different outcomes can result from the same initial conditions.

B.Steady-state or dynamic

Steady-state models use equations defining the relationships between elements of the modeled system and attempt to find a state in which the system is in equilibrium. Such models are often used in simulating physical systems, as a simpler modeling case before dynamic simulation is attempted.

Dynamic simulations model changes in a system in response to (usually changing) input signals.

C. Continuous or discrete

In **discrete models**, the state variables change only at a countable number of time points that mark execution of events and, thus, changes of state.

In **continuous models**, state variables change in a continuous way, and therefore there is an infinite number of states.

Examples of kinds of models include the following:

- A **continuous dynamic simulation** performs numerical solution of differential equations. Periodically, the simulation program solves all the equations, and uses the numbers to change the state and output of the simulation. Applications include flight simulators, racing-car games, and simulations of electrical circuits.
- A **discrete event simulation** manages events in time. Most computer simulations are of this type. In this type of simulation, the simulator maintains a queue of events sorted by the simulated time they should occur. The simulator reads the queue and triggers new events as each event is processed. It is not important to execute the simulation in real time. It's often more important to be able to access the data produced by the simulation, to discover logic defects in the design, or the sequence of events.

- **Agent-based simulation** is a special type of discrete simulation, which does not rely on a model with an underlying equation but can nonetheless be represented formally. In agent-based simulation, the individual entities (such as molecules, cells, trees or consumers) in the model are represented directly (rather than by their density or concentration) and possess an internal *state* and set of behaviors or *rules* which determine how the agent's state is updated from one time-step to the next.

Our approach to modeling is **agent-based, dynamic** and **discrete**. We believe that this approach is the best fit for creating an artificial world in which the agents feature a complex combination of capabilities – simulation of a physical and physiological organism, goal- and plan-based cognitive reasoning, communication between an artificial agent and a human and communication among artificial agents.

Is our model data-driven?

At this time, our model is primarily knowledge-driven, in the sense that it relies on symbolic modeling of causal chains of physiological, pathological and clinical events. Our approach proceeds from the assumption that the experts have created in their minds at least partial models of the above types of events, so that the scientific task is twofold:

1. eliciting from the experts the models that they have and formulate them explicitly in a descriptive theoretical statement;
2. validating (or, if you will, falsifying) these models empirically.

In order to facilitate the second task, one must

1. develop a formal computer simulation of the physiological, pathological and clinical events at the core of both the expert model and the formal theory that reflects it;

2. run the simulation on a representative sample of disease manifestations and either

   3a. have physicians judge whether the simulation is realistic or

   3b. compare the results of the simulation with actual medical records of instances of the diseases in question, their progression and treatment outcomes.

We plan to use both methods of validation/falsification. We have already started eliciting physicians' judgments about the system and plan to start using medical records as soon as possible. Using the medical records for validation is a classical option for making the model data-driven.

Are we using stochastic methods in our modeling?

Stochastic methods are used in our modeling but not as the only or even primary means of modeling. In our initial implementation, due to the nature of the application in which we decided to embody our model, it was essential to avoid randomness in the behavior of the artificial intelligent agent (the virtual patient) for pedagogical purposes – the teacher created patients that behaved in ways that he or she thought are the most relevant and efficient in the educational process. Note that different teachers were able to create different sets of sample virtual patients. Another reason for trying to eliminate randomness was to attempt to provide an even level field for examinees.

With all that, randomness and, therefore, the use of stochastic approaches, is present in the model in several manifestations. First of all, wherever clinical "bridges" were used in describing physiological and pathological processes whose mechanisms are not yet known to science, the clinical expertise of the expert physicians was elicited, formalized and encoded in the model. This expertise is formalized in the model largely through the use of value ranges (instead of single values) of a variety of model parameters (features). Selecting a value from such ranges for a particular instance of a virtual patient was the core operation during authoring virtual patient instances in our initial pedagogical application. To date, the elicitation process has not included judgments of distribution of values within the posited range. But for an application in which the authoring of patients is carried out automatically this stochastic capability can, and will, be added. One way of deriving knowledge for such distribution is stochastic and based on a large-scale analysis of medical records. Such a study is in its planning stages. Also note that expanding the team of expert physicians taking part in this process will inevitably lead to enhancing variability in the descriptions of the physiological and pathological properties, which will introduce even more potential randomness in the model.

Stochastic modeling is also used in several of the modules of the natural language processing component of our model, in cases where encoded knowledge fails to produce complete results.

Is our modeling approach agent-, team- or population-based?

In the classical use of these terms, population-based modeling is typically carried out stochastically and purely empirically. Agent-based modeling is often defined as concentrating on modeling individuals and can be done purely empirically or based on internalized knowledge. In practice, most agent-based approaches model agents that are rather simple and carry out a limited set of actions (methods). For example, a thermometer may be modeled as an agent capable only of measuring temperature.

Our approach does not reject this interpretation of the agent metaphor out of hand. However, we are centrally interested in building causal models of complex whose capabilities in some sense approach or model human capabilities. That is why our model can be called two-level agent-based model: it includes both low-level (e.g., thermometer-like) and high-level (decision-making) agents. In addition to the above, our interest extends beyond individual agents and to societies (teams) of agents. This is why our high-level agents must be endowed not only with decision-making capabilities vis-à-vis the world (including their own bodies) but also communication capabilities and capabilities of making decisions taking into account the fact that they are members of an agent society. Finally, as the agent societies we are interested in involve both artificial and human agents, it is essential to endow our artificial agents with the capability of communicating in natural language. This latter capability includes both understanding and generating text.

Unlike the majority of recent research into agent networks, our concern is not in allocating resources to collaborating uniform-skill agents who are working toward a common goal, our concern is to manage the interactions of “specialist” agents who must establish their own plans and goals and interpret those of other network agents.

Why do we believe that our work is feasible?

First, our development efforts are targeted toward **specific applications**: there is no attempt to develop a fully generalized, plug-in ready cognitive architecture (like TRAINS/TRIPS), or to implement a broad-coverage, domain-independent dialogue system, or to equip system agents with all of the plans and goals of human beings, or to endow them with the full spectrum of possible character traits (as is done in theoretical approaches to affective modeling), or to model diseases at a grain size any finer than that needed to support the given application. Instead, theoretical and practical advancements are geared toward the near- and long-term future of the specific systems, with infrastructure decisions being made with a long-term view but knowledge support targeted at near-term goals.

Second, the **integrated approach to knowledge modeling** in MVP permits the same ontological substrate to be used for knowledge-based simulation, planning, and NLP, meaning that once knowledge is encoded it is available to all system agents and processors. The OntoSem ontology used in MVP already includes over 8000 concepts, described by an average of 16 properties each; around 7000 of those are from the general domain, with the remaining 1000 devoted to medicine. Moreover, since scripts describing complex physiological and cognitive events are formally part of the ontology, the same scripting language used for physiological simulation (which is already understood by our simulator engine) can be used for planning and dialogue.

Third, the dialogue processing model is grounded in the **OntoSem deep semantic natural language processing system,** which has been under development for over 20 years.

Fourth, the past decade has produced a valuable **body of research** on cognitive engineering, agent networks, planning, plan- and goal-centered dialogue systems, etc. This large body of work includes inventories of needs for intelligent systems, sample architectures, descriptions of problems encountered, bridges between descriptive, theoretical and implementational work, and reports from the field that provide a good understanding of the current state of the art. In short, this body of work is permitting us to quickly reap the benefits of hard-won insights.

How can we validate our models?

There are three main methods of evaluation and validation of our approach:

- **Validation through use of an application system**. In the case of MVP, experts will manage virtual patients over time and evaluate whether patient responses and outcomes are in keeping with what the experts encounter in clinical practice.
- **Validation using an omniscient view of the system**. In the case of MVP, experts will manage patients while having access to the physiological properties that are used by the simulation but are unavailable to the user in a typical training scenario. The experts will evaluate whether the model encoded in the system is compatible with their own mental models.
- **Validation through comparison of real patient records with our models**. We could semantically analyze patient records, formulate their information in terms of our disease and treatment models, and see if our models cover the actual disease manifestations and physician decisions regarding when and how to treat.

How can our theory be falsified?

A theory is scientific only when it is falsifiable. We believe that our theory of modeling virtual patients is indeed scientific. Our theory is two-fold. One facet of the theory is the actual physiological and clinical knowledge encoded in the system. The other facet of the theory consists of the cognitive models of the physiological and reasoning agents.

The former facet of the theory can be falsified if expert physicians determine that a) the progression of diseases in the virtual patient does not correspond to their experience with such diseases; b) the reaction of the virtual patient to treatments does not correspond to their experience; or c) the treatments and diagnostics selected for a particular case or disease are not deemed appropriate. The cognitive model can be falsified if the behavior of the system agents is judged incoherent or unexplainable. (This behavior includes natural language dialog capabilities.)

Method and Application Comparisons

How does our work differ from data mining?

Data mining is primarily data-driven, with relatively little encoded knowledge brought to bear on the process of extracting useful “nuggets” of information from very large information streams. Often data mining concentrates on what is known as metadata, data about information (e.g., the author of a text, the date of its publication, the source where it was published, etc.). More advanced applications of data mining involve some analysis of the data extracted, described as “knowledge discovery.” Much of data mining uses databases as input. The core methods of data mining are empirical and statistics-oriented, though they can use prerecorded knowledge. We expect to use a variety of data mining techniques in the future, when we have collected a sufficient substrate of static knowledge and text processing capabilities to allow semantics-enhanced data mining over open text.

How does our work differ from expert systems?

Many of the AI systems in medicine have been expert systems, defined as systems that stand in for an expert, most typically by offering diagnostic assistance (for a list of over 50 such systems, see http://www.computer.privateweb.at/judith/index.html). Expert systems are, as a rule, not grounded in simulation and do not provide for extensive interaction. MVP is not an expert system, as it does not stand in for a physician. Furthermore, MVP does not follow the conceptual or architectural lead of classic AI systems, which were typically large and difficult to maintain sets of production rules. Instead, MVP has a proportionally far smaller inventory of rules that are more structured and aggregated, and contain control information that boosts system efficiency.

How does our agent society differ from most agent organizations?

A large number of different types of organization are in use in the area of multi-agent systems. Horling and Lesser (2005) distinguish hierarchies, holarchies, coalitions, teams, congregations, societies, federations, markets, matrix organizations and compound organizations. Our approach uses some features of agent team and some of agent society. Horling and Lesser define an agent team as follows: “An agent *team* consists of a number of cooperative agents which have agreed to work together toward a common goal. In comparison to coalitions, teams attempt to maximize the utility of the team (goal) itself, rather than that of the individual members. Agents are expected to coordinate in some fashion such that their individual actions are consistent with and supportive of the team's goal. Within a team, the type and pattern of interactions can be quite arbitrary but in general each agent will take on one or more roles needed to address the subtasks required by the team goal. Those roles may change over time in response to planned or unplanned events, while the high-level goal itself usually remains relatively consistent.” The above, of course, is applicable, under some interpretation, to most cooperative multi-agent systems. Agent societies are understood as a looser organization, with a “characteristic set of constraints they impose on the behavior of the agents, commonly known as *social laws*, *norms* or *conventions*. These are rules or guidelines by which agents must act, which provides a level of consistency of behavior and interface intended to facilitate coexistence.”

There is, however, a big difference between our environment and the environments described in Horling and Lesser: the environments of the kind we build, by definition, involves a human as one of the agents.

How do our models differ from those based on canned scenarios?

Most currently available clinical decision-making systems are not grounded in simulations. Instead, they provide trainees with the opportunity to work through decision trees that target key decision points in the process of diagnosing and treating a disease. MVP, by contrast, offers trainees a more open-ended choice space and a richer scope of interactions, which more closely parallel the demands of actual medical practice.

What are our differences with other projects/systems?

CIRCSIM

CIRCSIM-Tutor is a system whose focus has fundamentally shifted since its inception (Evens and Michael 2006). Initially, under the name MacMan, the system was a mathematical model of the baroreceptor reflex that could be explored by students but provided no feedback. However, it was found that the lack of feedback made it a non-optimal teaching tool, which led to the development of the first spin-off system, Heartsim, which offered limited feedback. With the Heartsim system, developers realized that the mathematical model was not being exploited and that the most effective teaching was based on stored correct predictions rather than real-time calculations using the mathematical model. As such, the final system, CIRCSIM-Tutor (which is still under development), does not actively use the mathematical model: the dynamic aspect of the system is constrained to the tutoring process itself. To summarize, this system went from offering students a dynamic mathematical model with no tutoring support to offering them tutoring without the dynamic mathematical model. In the MVP system, by contrast, the autonomous functioning of MVPs is, and will remain, no less important than the mentoring aspect of the application.

The developers of CIRCIM are pessimistic about the prospects of automatic tutoring in a less than highly constrained realm:

> “When we started the CIRCSIM-Tutor project 15 years ago, some experts in the field argued that student modeling was too difficult to be worth the trouble; some even classified the problem as totally intractable… Anyone who observes human tutors in action, on the other hand, must recognize that they base decisions at all levels, from the choice of the next problem to present to the student to what kind of hint to provide, on their model of the student… Joel Michael and Allen Rovick were so convinced of the crucial importance of modeling that they picked the CIRCSIM domain [the baroreceptor reflex] for our tutor largely because they felt that it would be easy to construct a good student model in this subject area … They are… convinced that it is important to build a comprehensive model before starting to tutor, to ensure that the tutor can begin by attacking the most important of the student’s conceptual difficulties” (p. Evens and Michel 2006: 252-3)

Undoubtedly, selecting a narrow purview facilitates domain modeling, student modeling and the automation of tutoring support; and, all other things being equal, one would expect better near-term results from a more highly constrained system. However, all other things never really are equal: there is a real-world need for simulation and tutoring in the broad domain of diagnosis and treatment of disease, and it is this need that has set the agenda for our research and development. While task-driven projects necessarily involve unknowns, they also promise exciting new horizons both within the targeted application and beyond it.

TRAINS/TRIPS

TRAINS and its successor TRIPS (hereafter, TT) are projects devoted to the study of natural language dialogue between an intelligent agent and a human as they collaborate on a planning task – specifically, cargo transport. The interconnection between dialogue and planning is key in these projects – as it is in MVP. Of particular interest for us is the work of David Traum, who has created a bridge from theoretical linguistic descriptions of dialogue to practical implementations of dialogue systems within a plan-oriented cognitive architecture. We are not, however, attempting to implement a system just like TT. In fact, there are at least four differences between the two environments: 1) TT is based on a generalized architecture that must manage compatibility issues between plugged in components, whereas the architecture of MVP is system specific, thus reducing one type of complexity; 2) TT does not pursue the depth of semantic text processing that MVP does; 3) TT does not incorporate a sophisticated simulation system; 4) TT does not include a tutor (the domain does not call for one). By having a broader scope of work devoted to a more complex domain, MVP clearly involves more risk, but also promises a bigger payoff: a training and mentoring system to support the next generation of physicians. For further information, see the project website: http://www.cs.rochester.edu/research/trains/

Knowledge Machine

KM is a knowledge representation and reasoning system developed by a group at the University of Texas. It permits users to encode knowledge, query the database about the knowledge, and run simulations that would indicate what the state of objects in the given world would be if certain events took place.

KM and OntoSem are both frame-based knowledge representation languages, the former focusing on logic-based reasoning and the latter on text processing and, as of late, simulation. Some points of similarity among the environments are:

- a three-tiered distinction between concepts (called ‘classes’ in KM), instances, and a hybrid entity that lies somewhere in between (in OntoSem this hybrid is called an ontological instance; in KM it is called a proto-instance). While the definitions and distribution of these three types of entities is not precisely parallel in the two environments, the theoretical and methodological reasons for making such distinctions overlap

- the use of primarily first-order logic in describing entities (concepts, instances and those hybrid entities) using properties and fillers
- the possibility of placing complex fillers in slots, nesting such descriptions to any depth, and tracking coreferences within such nested structures (permitting the construction of scripts and prototypes)
- the use of a large inventory of properties, and the possibility of adding more as an acquirer deems necessary
- the support of knowledge expressed in if-then statements
- the support of reasoning for sophisticated applications.

The main difference between KM and OntoSem involves the focus of development effort and the applications supported. KM is a freely distributed environment that permits users to created databases and reason over them using theorems of predicate logic. OntoSem, by contrast, focuses on natural language processing in general as well as simulation in the medical domain.

Other Knowledge-Based Simulation Work in Medicine

Recent literature includes two projects that offer interesting points of comparison with the MVP system. Walton Sumner and his associates (Sumner and Hagen 2006, Sumner et al. 1996, 1996, Marek et al. 1996) developed a system for the purpose of advancing medical certification procedures of the American Board of Internal Medicine beyond the level of multiple-choice questions. The system relies on simulating a patient with possibly multiple co-existing conditions and allowing the examinee to intervene. The system addresses the issue of automatically creating instances of virtual patients by starting with a selected disease (or diseases) and probabilistically creating a medical history for each patient instance based on knowledge about the incidence of this disease (or diseases) in the population. This emphasis on generating differentiated patient histories is due to the perceived need for providing a secure testing environment in which patient instances are not reused. While the knowledge in the system covers "health states" of a virtual patient and causal and temporal connections among them (with their associated properties and symptoms), this knowledge does not yet support a realistic simulation of a virtual patient's physiological processes. The probabilistic nature of much of the operation of this system suggests the use of Bayesian networks as the underlying representational mechanism, which adds complexity to both knowledge acquisition and processing.

Of the many differences between this system and the MVP system, we believe the most salient are: a) the Sumner system focuses only on evaluation, whereas MVP centrally includes training; b) MVP provides for following patients over time without the need for any explicit intervention; c) MVP achieves a level of realism in patient simulation not attempted by the Sumner system, particularly by virtue of endowing the VPs with a cognitive agent capable of perception (sensory and language), reasoning (goal-oriented decision making) and action (physical and verbal); d) MVP permits the user to participate in an agent network that involves other human-like agents, including a dedicated tutoring agent; e) MVP uses stochastic methods much less than the Sumner system in order to keep all data and processing explicitly controlled and inspectable by developers.

Amigoni et al. (2003) describe a multiagent system for the modeling and regulation of physiological phenomena, specifically, for regulating the insulin and glucose levels in diabetes patients. This system relies on what the authors call the "anthropic agency" architecture, "a powerful paradigm to develop control systems for physiological processes shaped as multiagent systems." (p. 310). The architecture consists of the traditional steps of perception (called "knowledge extraction", implemented using a team of identical "extractor agents"), reasoning ("decision making", implemented using a team of identical "decisional agents") and action ("plan generation" with their team of "actuator agents"). Communication among agents from different groups is mediated through messages on blackboards. Communication among agents of the same group is prohibited. The blackboards are serviced by their own agents, and the overall architecture also includes a database agent and "majordomo" agent responsible for communication with "both the technical expert, who can modify the composition of the system by adding and removing agents, and the medical expert, who can inspect and tune the parameters and the functioning of the system." (p. 313). The actual implementation involves one extractor, one actuator and two decisional agents and essentially simulates the fluctuation of just two properties – glucose and insulin levels with two outside influences – food intake and physical activity.

All the agents in this system are what we call low-level agents. The physiological model has a narrowly directed coverage, no cognitive abilities are simulated for the virtual patient and no network of high-level agents simulating human capabilities is introduced. In general, the complexity of the domain knowledge is not the main focus of this work. The main thrust of the paper is the discussion of the algorithms that embody the agents and the algorithms supporting their communication (the negotiation mechanism). This is understandable considering that the intended application of the anthropic agency is intelligent prosthetics – implanting an intelligent insulin supply regulator in diabetic patients.

Our work, by contrast, is devoted to immersing human users in simulated environments. Therefore, we concentrate on the breadth of coverage and realism of the simulation and postpone the discussion of such architectural and control issues as resource allocation, efficiency of agent collaboration and task scheduling. This deferral is made possible by making a variety of simplifying assumptions in our system. For example, we assume that the scheduler, executor and the demons have all the actual time they need to complete all their operations at a given agenda time slot. In other words, virtual time is not equated with real time. Our simulations at this time do not suffer because of this assumption because we have been dealing with chronic diseases that unfold over long periods of time. We will revisit this simplifying assumption when we will start modeling acute diseases where durations of crucial events can be very short. Similarly, our application environment allows the simplifying assumption of full availability of not only computational resources but also resources in the world being modeled. By the same token, our system's quality and utility will not suffer if we do not concentrate on scheduling particular specialists or lab technicians for particular tasks. Since the one agent in our system that is intended to operate under uncertainty is the human user, we can avoid the complexity of introducing probabilistic reasoning engines.

Should the need for removing the simplifying assumptions arise in our future work, we intend to use advances in multiagent coordination, which is a very active area of research (cf., e.g., Lesser et al. 2004 or Pynadath and Tambe 2003). Our work is complementary to such studies. Using multiagent coordination results, we can improve the efficiency of our systems, while the coordination testbeds will benefit from being able to test the coordination algorithms and architectures using the knowledge-rich and heterogeneous agents developed in the MVP system.

Other System Comparisons

Several existing systems and projects share a small number of characteristics with our approach. In what follows, we give a very brief survey of such systems.

Realistic Simulation. One type of computer-aided training involves technical task trainers. These focus on training a specific technical step, with little or no cognitive simulation. Like our environment, they aim to be sophisticated and lifelike. For example, manikins to teach the care of infants and adults have been developed by Laerdal, Inc. ("SimBaby", http://www.laerdal.com/; "SimBaby") and Meti, Inc. ("The Human Patient Simulator", http://www.meti.com/), respectively.

Cognitive Training. Among the computer methods to train decision-making skills are systems based on decision trees that embody diagnostic and treatment algorithms at the case level. These include no biomechanistic processes, and the user is limited to selecting one of the pre-scripted options at fixed points in case. MedCases, Inc. (http://www.medcases.com) is an e-learning company that develops patient interaction scenarios for continuing medical education. Although these are far more "canned" that the ones in MVP, they too seek to train cognitive capabilities.

Hybrid Models. A well-known simulation project is the Virtual Soldier (http://www.virtualsoldier.net/), which simulates the human thorax in the context of penetrating trauma. It combines the Foundational Model of Anatomy with a stochastic physiological knowledge at the cell, tissue, body and population levels. Although Virtual Soldier differs from our work in several ways – as by focusing on the short-term treatment of trauma rather than the long-term treatment of patients with ever-changing disease states – the approach integrates various types of knowledge, as does ours.

Intelligent Behavior in Changing Circumstances. Like the well-known expert systems (e.g., Mycin Shortliffe XXX), MVP shows intelligent behavior in ever changing circumstances. However, unlike traditional expert systems, our system is grounded in simulation, stresses language-based interaction, uses the same knowledge bases for both simulation and interaction, and permits free-form interventions – all of which involve innovative uses of adaptive computing.

Large Population of Patients. This is a feature which is shared by our approach with that of Sumner and Hagen (2006).

Medical Tutoring. The CIRCSIM project (Evens and Michael 2006) concentrates on tutoring in a medical domain and involves natural language dialog – just like MVP. However, CIRCSIM-Tutor currently does not incorporate simulation, and it covers only one specific medical condition, the baroreceptor reflex – the body's rapid response system for dealing with changes in blood pressure.

Knowledge-Based Dialog. Susan McRoy at the University of Wisconsin, Milwaukee has been developing a dialog system in the framework of a tutoring environment for medical students (e.g., McRoy et al. 1997). She shares our belief in the need for knowledge for language processing but focuses on dialogue issues without detailed specification of physiological and pathological states.

Appendix C:

A Ubiquitous Context-Aware Environment for Surgical Training

P. Ordóñez, P. Kodeswaran, V. Korolev, W. Li, O. Walavalkar, B. Elgamil, A. Joshi, T. Finin, Y. Yesha
Department of Computer Science and Electrical Engineering
University of Maryland, Baltimore County
Baltimore, MD USA
{ordopa1, palanik1, vkorol1, wenjia1, onkar1, ben8, joshi, finin, yeyesha}@cs.umbc.edu

I. George
Department of General Surgery
University of Maryland Medical Center
Baltimore, MD USA
igeorge@smail.umaryland.edu

*Abstract***— The age of technology has changed the way that surgeons are being trained. Traditional methodologies for training can include lecturing, shadowing, apprenticing, and developing skills within live clinical situations. Computerized tools which simulate surgical procedures and/or experiences can allow for "virtual" experiences to enhance the traditional training procedures that can dramatically improve upon the older methods. However, such systems do not to adapt to the training context. We describe a ubiquitous computing system that tracks low-level events in the surgical training room (e.g. student locations, lessons completed, learning tasks assigned, and performance metrics) and from these derive the training context. This can be used to create an adaptive training system.**

Keywords- context awareness; ubiquitous computing; surgical training.

I. INTRODUCTION

Context aware ubiquitous computing systems must process streams of data arriving from sensors, services, devices and other systems to construct and maintain a model of their environment. If the environment is complex, the volume of data will be large and if the system aspires to be intelligent, the processing over the data may be computationally expensive. In ongoing research, we are designing and implementing a framework for constructing intelligent, context-aware ubiquitous computing systems.

We are pursuing the general technical goals while working with colleagues at the University of Maryland Medical Center (UMMC) to use an evolving project to implement a system named CAST, Context-Aware Surgical Training. CAST is part of the Operating Room of the Future (ORF) [15] project that is housed in the newly opened Maryland Advanced Simulation, Training, Research, and Innovation Center (MASTRI). It is a facility with authentic operating rooms specially renovated/constructed and instrumented to support innovative research and training. We have already constructed and deployed a partial prototype of the CAST system in the MASTRI Center to test the feasibility of our approach, which is described in this paper.

II. THE CAST VISION: BACKGROUND AND MOTIVATION

Traditionally, surgical training has consisted of the resident shadowing senior surgeons and practicing diagnostic and procedural skills on live patients. In 1999, Gorman et al. [4] stated that estimated costs of training chief residents for general surgery alone cost $53 million per year. A long standing debate over the ethics and practicality of such practices is also of concern [2]. Furthermore, statistics like the following demonstrate the need for a dramatic change in clinical pedagogy. A survey of residents and faculty in surgical training programs in 2003 described that more than 87% of the 1,653 responses from residents surveyed indicated that they had an 80 hour work week. 45% reported working more than 100 hours per week. 57% reported that their cognitive abilities had been impaired by fatigue [5]. Furthermore, although apprenticeships have been shown to be very effective, in the case of a surgical procedure, the well being of a patient outweighs the training of the resident.

Computer-enhanced simulations show promise for addressing all of these concerns. However, as Granger [7] states in his dissertation, "The key issue is not whether to creep forward through evolution of digital substitutes, but whether to promote the revolution of clinical practice through the integration of pervasive computing technologies."

Our system aims to improve the training provided by simulators by making them a part of a context-aware training environment. This allows the training process to require less direct intervention from mentors in many of the routine tasks. We aim to reason over sensed data streams to infer context about the events in the training process. In the initial prototype system described in this paper, we focus on laparoscopic surgical training[1]. We track the presence of surgical residents in the training rooms, which training machine they have used and for how long, which lessons they have downloaded etc. This information is then used to guide the students to the practice/tests they need to take. Similarly, recordings of students' hands as they use the laparoscopic trainers as well as the output from the simulators are made available to instructors, who can then view and analyze them, add comments and annotations, and suggest skills on which the

This work is supported in part by award W81XWH-06-2-0057 from USAMRMC/TATRC. The views/opinions/findings are those of the authors alone and not the supporting agency.

[1] Laparoscopic surgery involves operations in the abdomen that are performed through small incisions rather than the larger ones required by traditional surgical procedures.

trainee needs to focus. Instructor feedback and suggestions can be automatically provided to trainees as podcasts or text messages.

An Electronic Student Record (ESR) provides a centralized repository of student information and progress, and helps infer their appropriate pedagogic context. This record is described in the semantic web language RDF-S, but is presently implemented as a database schema. The ESR will provide a comprehensive summary of the students' progress such as the time spent at each machine, chapters checked out, video captures etc. which will help the instructor to review student performance without physically being present in the training room.

In this system, the student benefits from the guiding elements that can be brought to bear and the real time adaptation of the training. Moreover, the trainer is now able to change their curriculum to meet the needs of their students. For the patient, the movement to disrupt the old and oft-repeated mantra of "See one, do one, teach one" is quite telling. In particular, the steep learning curves of new surgical technologies can now be mastered by trainees outside of live operative settings.

III. Related work

"William Osler wrote in 'The Principles and Practice of Medicine' in 1982 that: 'To learn medicine without books is to sail an uncharted sea, while to learn medicine only from books, is not to go to sea at all.'" [7]. In the 21st century, the question is can we virtually go to sea?

A. *Virtual Reality Training*

Parallels have been drawn between pilots and surgeons in that both must be able to respond to potentially life-threatening situations in unpredictable environments [4]. A pilot must be prepared to land a plane when several engines have failed and a surgeon must be able to respond to a cardiac arrest in the middle of open heart surgery. Flight simulators have long been used to train pilots for the worst of circumstances. In fact, the simulators of today are so effective that they are often used to train a pilot on a new version of a plane, and the pilot flies the real plane on a scheduled flight [10].

As a result, surgical simulation is rapidly becoming the standard for surgical training. Training simulations currently exist for endoscopic sinus surgery [3], ossiculoplasty surgery [8], and orthopedic surgery [7] to name a few. Many of these simulations create a virtual reality using video gaming technology. A recent paper in 2007 correlated video gaming skills with the laparoscopic surgical skills [12], although that should not be a reason to relax concerns about the amount of time children spend playing video games.

Some of the aforementioned systems use multimedia and hypermedia to enhance surgical training [7]. Others simply use 2-D video and haptic devices as in most video games [3][8]. Others use a hybrid approach where they combine the 2-D video with a visual awareness of objects and events in a room [11]. Welch et al. are capturing and displaying high-fidelity 3-D graphical reconstructions of the actual time-varying events for the purpose of doing on-line consultation and off-line surgical training [9]. This research could help to provide surgical training and mentoring by specialists to generalist doctors in isolated hospitals in developing countries [6].

The 3-D graphical reconstructions are being stored in Immersive Electronic Books (IEB) for surgical training. Via IEB surgeons can explore previous surgical treatments in 3-D [10]. Thus, in the same way, a pilot can test out a new version of a plane time and time before she flies it, a surgeon can see a surgical procedure and interact with it time and time again until she performs it.

B. *Other Training*

B-line Medical [1] provides what it describes as a Clinical Skills System that is a comprehensive digital solution for managing and operating a clinical skills center. The system has four major components: user management and content creation, exam management, scoring and reporting, and professional quality audio/video. This system attempts to address the same concerns about efficiency and automation in a surgical training environment. It uses a card swiping mechanism to identify residents and monitor their progress. It is mostly built on existing content management technology, and is not concerned with inferring context from sensed data.

More generally, to the best of our knowledge, none of the existing surgical training systems seek to infer significant events from the sensed data, or to use such data to infer the context of the surgical procedure and create a smart , adaptive surgical training space.

IV. Overall Architecture

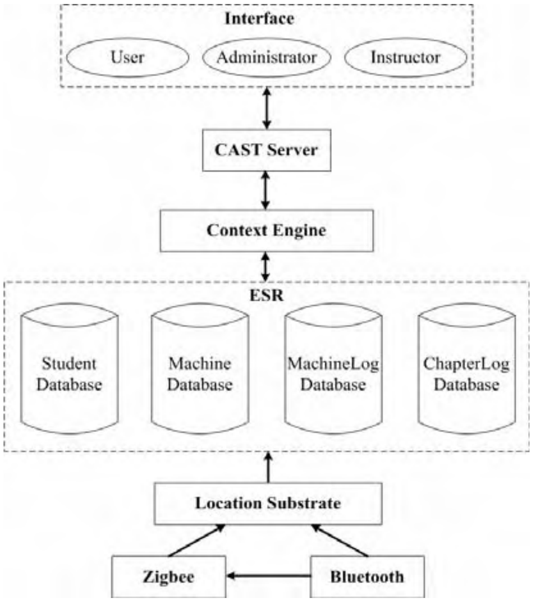


Figure 1. A high-level overview of the CAST system.

As in any ubiquitous computing system, location plays an important role in CAST. As shown in Figure 1, we use a location substrate consisting of a combination of Zigbee and Bluetooth to provide location information. This location information is then fed into a location database which keeps track of information such as which students were in front of a simulator and for how much time. The simulators in general require that students complete relevant chapters from the Fundamentals of Laparoscopic Surgery (FLS) training program before working on them. We host the FLS chapters on a web server that students can access through their logins. The chapters checked out are stored as part of the student's Electronic Student Record (ESR), which is updated when students check out chapters through the web interface. When the location substrate detects that a student is standing next to a simulator, it queries the student's context to verify that the student has completed the required FLS chapters. Only if the student has finished the required chapters is he/she allowed to work on the simulator.

Currently, information from sensors, such as training boxes, video recorders, RF tags, and cell phones, provide basic context information. These low level data streams are processed to generate higher-level primitive events, such as a resident entering the training room. A hierarchical knowledge-based event detection system correlates primitive events, resident data, and workflow data to infer high-level events, such as the finishing of a training module. Video streams of the training procedure are time-stamped and labeled with the inferred higher-level events. . These video recordings, location data, and performance data from simulators can be viewed offline by instructors. Moreover some simulator manufacturers are working on providing automated performance evaluation through video metric analysis. This resulting analysis, where available, could also be used as part of the ESR. The resulting ESR will provide trainers of physicians with a permanent record of the training session including an evaluation of the trainee's performance, the duration of the session, the number of times the trainee attempted the module before attempting the exam, and a labeled and time-stamped video of the session.

In a hospital setting with 10-20 residents, the smart training space will monitor the training activities of each resident more closely, and improve the workflow in the training center by allowing residents to sign up for a simulator at an allotted time only if they had the appropriate prerequisite tests/lessons and had been cleared by their mentor. Thus, a trainee surgeon may practice at a simulator during hours of convenience and be evaluated at the end of a session without the need of a trainer.

More generally, an important contribution of our system is that it makes the entire process of surgical training asynchronous. The instructor no longer needs to be physically present with the student during training. Many of the "adaptations" that a physically present instructor would have made (guiding the students to the right simulators and pointing out particular skills they needed to master, for instance) are now done by the system automatically. Moreover, the capture of the data stream from the simulators and the video of the trainee's hands as he/she practice on the laparoscopic simulators allow the evaluation to be done separately as well. This can also help remove the location dependence of surgical training. For instance, a student could do his training at any available simulation center (as long as it is networked with the parent school) and still have his/her procedure reviewed by the instructor back in the parent medical school, or even by an instructor at a different school.

CAST will also alleviate the burden of viewing the entire training video for evaluation even where automatic video metric analysis is not available or possible. It will provide the trainer with a labeled and time-stamped video of each training session that is correlated with the events signaled by the underlying simulator. For instance, the simulator may signal that the cut was made outside the designated area. Since the video timestamps will be correlated to the timestamps of the simulator output, the trainer can jump to portions of the video which are critical.

V. Localization

Based on experiments conducted at the MASTRI, the Awarepoint^{TM} system, like most other commercially available location systems such as Ekahau [22], provides room level accuracy which suits the typical requirements of a hospital for asset or personnel tracking. However the CAST system requires finer localization to be able to place a student as being in a position to operate a particular simulator when there may be more than one in a room. We decided to use Bluetooth to provide localization at this granularity. As a result, our system uses a combination of Awarepoint and Bluetooth for localization.

A. *Awarepoint^{TM}*

Awarepoint seeks to address the limitations in RFID technology and claims to have developed a real-time solution for one of hospitals' major problems, tracking of the movement of their staff, patients, and equipment. Awarepoint's tracking system is based on Zigbee, a high level communication protocol using the IEEE 802.15.4 standard for wireless network [16]. It is designed for radio frequency applications which require a low data rate, long battery life and secure networking. Awarepoint base stations, plugged directly into wall sockets, form a mesh network to deliver data from tags (such as signal strength and identifier) to a server which then uses a proprietary approach to identify the location of the tag. Each trainee is assumed to carry a tag. Awarepoint's standard user interface is a GUI that shows the location of the tags in a facility map. However, this is not appropriate for our purpose. As a part of a collaborative effort with Awarepoint, we have been provided access to their server database and associated SOAP interfaces so that we can directly query the location of a tag.

B. *Bluetooth Module*

We use Bluetooth to provide machine-level location information so that instructors can query for information such as how much time students spend in front of a machine. We periodically broadcast a Bluetooth device inquiry message, which returns the devices in range which respond to the inquiry. However, this method has high latency and does not necessarily return all Bluetooth devices in range, as some of them may not be listening on the same channel as the inquiry was sent and hence may not respond. As a result we decided to use a different approach. Our approach is motivated by the fact that we are not looking for "any" device but only for devices belonging to trainees. Each trainee is assumed to always have a Bluetooth capable device on him/her and the device address-student association is maintained in the student database. In our method, we periodically initiate connections to a list of MAC addresses obtained from the student database, and if the connection succeeds, we can infer that the corresponding device, and hence student, is in range. This method works well when the number of students is small, but the time to discover a device in this case grows with the number of students. To reduce the number of MAC addresses to initiate connections to, we use the Zigbee location information which provides room level accuracy, and initiate connections only to devices belonging to trainees currently in the room. When a single trainee is near a simulator, this suffices. However, since multiple trainees could be in the range of a machine, we still need a way to distinguish which one is actually using the machine. We achieve this by displaying a drop down list of students in range and requiring students to log in before using the machine. Thus we provide an additional layer of authentication, when Bluetooth discovery alone is unable to identify a student.

VI. Fundamentals of Laparoscopic Surgery (FLS)

FLS is "a comprehensive, CD-ROM-based education module that includes a hands-on skills training component and assessment tool designed to teach the physiology, fundamental knowledge, and technical skills required in basic laparoscopic surgery. [14]." It was created by the Society of Gastrointestinal and Endoscopic Surgeons (SAGES) which is accredited by the Accreditation Council for Continuing Medical Education (A.C.C.M.E.) to sponsor Continuing Medical Education for physicians. To the best of our knowledge, FLS is the only CD-based education module that can be used to acquire CME credits. Since our system needs to incorporate the FLS curriculum and move away from its present CD based model, we need to obtain appropriate permissions. While this is being discussed, we have mocked up curriculum to represent the 14 modules in the FLS as shown in Figure 3.

A. *Webservice and MySql database*

We are hosting the mocked FLS curriculum on an Apache web server. We are using video clips to represent each of the modules. The user has to authenticate herself to the system to check out a training module. We track when students log in, for how long, which chapters they check out, in what sequence, and then log the information in the corresponding tables in the ESR. This information is useful in analyzing student progress. It also allows us to direct students to the tests they need to take and the procedures they need to practice on particular machines. So for instance if a trainee tries to use a simulator for which she has not checked out the appropriate lessons, we prevent her from using it.

VII. Implementation Details

The target development platform for CAST is the Nokia N800 [18]. The N800, pictured in Figure 2, has an impressive set of features such as Bluetooth, WiFi, and an inbuilt camera in a small form factor which makes it an attractive choice for capturing context in the training room. The N800 runs a Debian Linux distribution, which makes developing and porting applications easy. Each simulator has an associated N800 device. It serves as our Bluetooth location base stations. We use the built in camera to capture live video streams of students' training and relay it over WiFi to a central video database for indexing, potential automatic analysis, and review by the instructor. The N800 can also accept the simulator data feeds and stream them to the ESR. Our application code is mostly written in Python, using python for maemo [17]. N800's built in video chat, RSS feedreader, and podcast applications are also useful in allowing trainee-mentor interactions.

We have implemented Bluetooth localization using the PyBluez [19] module and BlueZ [20] stack on Linux. The Awarepoint server exports location information both through a database and a web service. We currently use the Awarepoint web service to obtain room level location information.

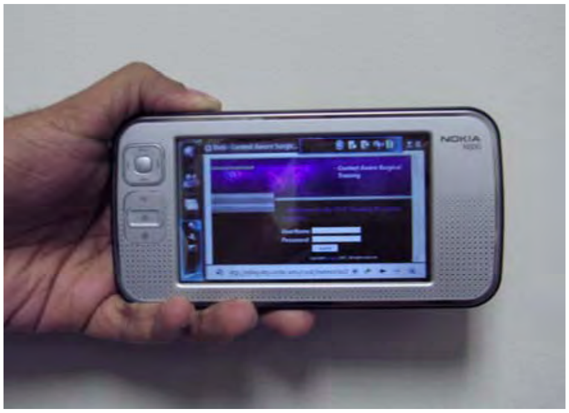
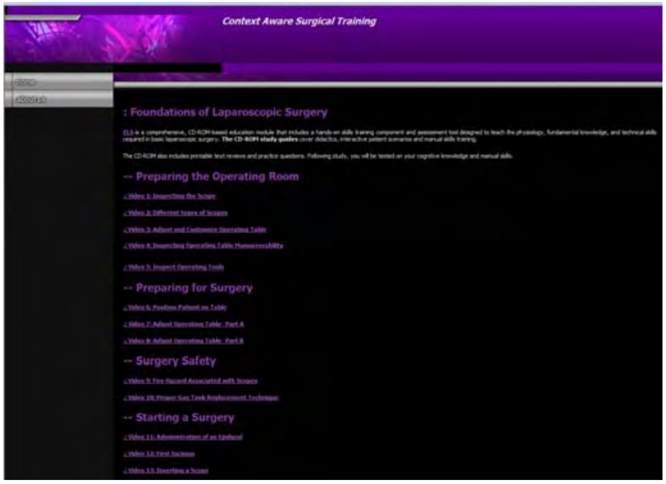


Figure 2. Login page to FLS Interface on Nokia N800.

The ESR is defined using RDF-S. Figure 4 gives a snapshot of the ESR in RDF-S. In the present implementation, we do not use a triple store. Instead, we have a preliminary version of the ESR as a MySQL database to allow for rapid prototyping. As we begin to reason over the sensed data more fully, we will migrate towards a triple store such as Jena [23].


Context Aware Surgical Training

: Foundations of Laparoscopic Surgery

FLS is a comprehensive, CD-ROM-based education module that includes a hands-on skills training component and assessment tool designed to teach the physiology, fundamental knowledge, and technical skills required in basic laparoscopic surgery. **The CD-ROM study guides** cover didactics, interactive patient scenarios and manual skills training.

The CD-ROM also includes printable text reviews and practice questions. Following study, you will be tested on your cognitive knowledge and manual skills.

-- Preparing the Operating Room

- Video 1: Inspecting the Scope
- Video 2: Different types of Scopes
- Video 3: Adjust and Customize Operating Table
- Video 4: Inspecting Operating Table Maneuverability
- Video 5: Inspect Operating Tools

-- Preparing for Surgery

- Video 6: Position Patient on Table
- Video 7: Adjust Operating Table - Part A
- Video 8: Adjust Operating Table - Part B

-- Surgery Safety

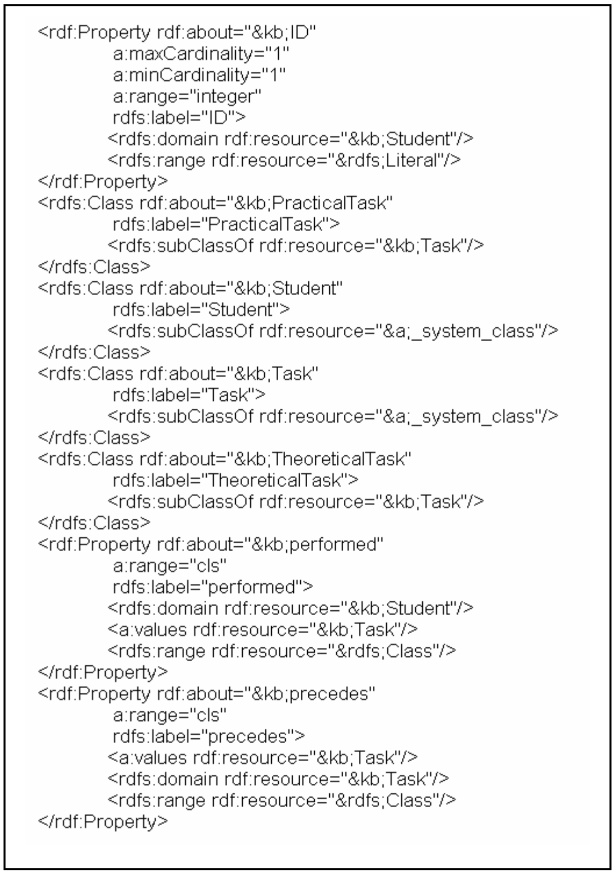
- Video 9: Fire Hazard Associated with Scopes
- Video 10: Proper Gas Tank Replacement Technique

-- Starting a Surgery

- Video 11: Administration of an Epidural
- Video 12: First Incision
- Video 13: Inserting a Scope

Figure 3. Mock FLS Curriculum

We have implemented and integrated the various modules of the system to create a prototype application. We can populate the student locations from the Awarepoint system, use Bluetooth to recognize trainees, and keep track of (mock) FLS modules checked out. We can also capture the video stream, although we do not yet synchronize it with the data stream captured from the simulators. This is because the simulator data streams are in proprietary formats for which no public documentation is available. We are presently working with some of the device manufacturers to understand their formats. One of the simulators used in the initial CAST deployment is shown in Figure 5.



```
<rdf:Property rdf:about="&kb;ID"
           a:maxCardinality="1"
           a:minCardinality="1"
           a:range="integer"
           rdfs:label="ID">
          <rdfs:domain rdf:resource="&kb;Student"/>
          <rdfs:range rdf:resource="&rdfs;Literal"/>
</rdf:Property>
<rdfs:Class rdf:about="&kb;PracticalTask"
           rdfs:label="PracticalTask">
          <rdfs:subClassOf rdf:resource="&kb;Task"/>
</rdfs:Class>
<rdfs:Class rdf:about="&kb;Student"
           rdfs:label="Student">
          <rdfs:subClassOf rdf:resource="&a;_system_class"/>
</rdfs:Class>
<rdfs:Class rdf:about="&kb;Task"
           rdfs:label="Task">
          <rdfs:subClassOf rdf:resource="&a;_system_class"/>
</rdfs:Class>
<rdfs:Class rdf:about="&kb;TheoreticalTask"
           rdfs:label="TheoreticalTask">
          <rdfs:subClassOf rdf:resource="&kb;Task"/>
</rdfs:Class>
<rdf:Property rdf:about="&kb;performed"
           a:range="cls"
           rdfs:label="performed">
          <rdfs:domain rdf:resource="&kb;Student"/>
          <a:values rdf:resource="&kb;Task"/>
          <rdfs:range rdf:resource="&rdfs;Class"/>
</rdf:Property>
<rdf:Property rdf:about="&kb;precedes"
           a:range="cls"
           rdfs:label="precedes">
          <a:values rdf:resource="&kb;Task"/>
          <rdfs:domain rdf:resource="&kb;Task"/>
          <rdfs:range rdf:resource="&rdfs;Class"/>
</rdf:Property>
```

Figure 4. Snapshot of ESR in RDF-S.

VIII. Future Work

In the near future, we will evaluate the impact of such a context aware, ubiquitous system for surgical training in collaboration with colleagues at the University of Maryland Medical System. We will consider the security, effectiveness and efficiency of the system. Other simulators at MASTRI include a ProMIS^{TM} surgical simulator as well as a METI^{TM} Human Patient Simulator^{TM}. The ProMIS^{TM} surgical simulator includes performance metrics that have been validated for use with the SAGES FLS program that we intend to use in the ESR. The METI^{TM} Human Patient Simulator^{TM} provides a log of vital signs during a surgical training procedure. We aim to expand CAST to capture data from these sources as well.

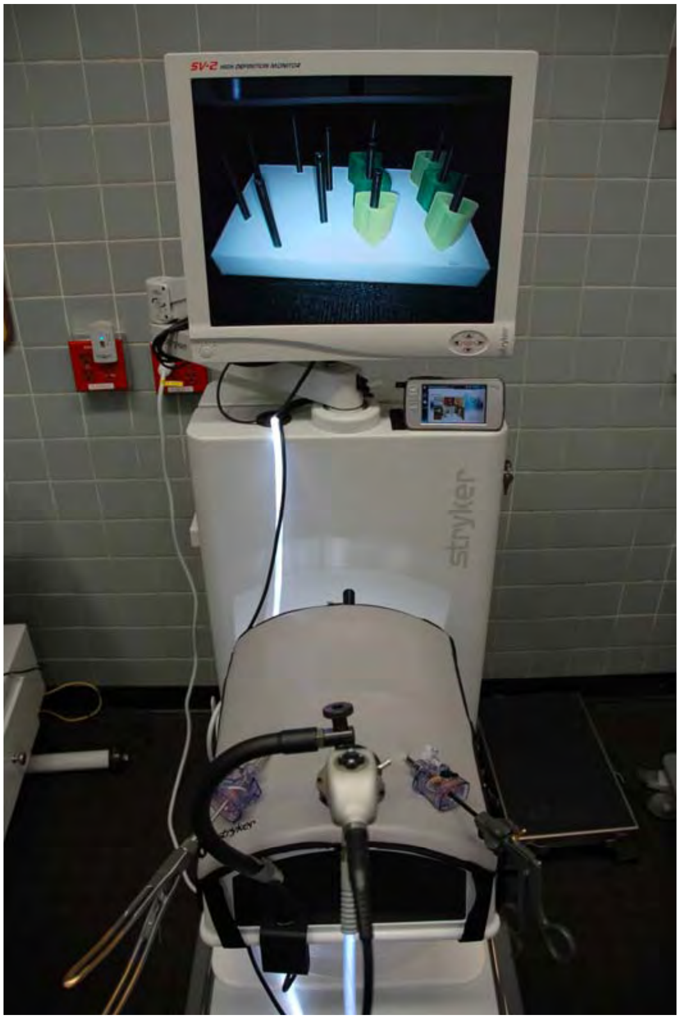


Figure 5. A Laparoscopic Training Simulator with an N800.

References

[1] "Clinical Skills System", B-Line Medical: Comprehensive Digital Solutions, http://www.blinemedical.com/solution/clinicalskills.

[2] M.A Reznek, P. Harter, and T. Krummel, "Virtual reality and simulation: training the future emergency physician," Acad. Emer. Med., vol. 9, issue 1, pp. 78-87.

[3] M. . Fried, R. Satava, S. Weghorst, A.G. Gallagher, C. Sasaki, D. Ross, M. Sinanan, J.I. Uribe, M. Zeltsan, H. Arora and H. Cuellar, "Identifying and reducing errors with surgical simulation," Qual. Saf. Health Care, vol. 13, pp. 19-26, 2004.

[4] P.J. Gorman, A.H. Meier, T.M. Krummel, "Simulation and virtual reality in surgical education: real or unreal?," Arch. Surg., vol. 134, pp. 1203-1208, Nov 1999.

[5] M.J. Niederee, J.L. Knudtson, M.C. Byrnes, S.D. Helmer, R. S. Smith, "A survey of residents and faculty regarding work hour limitations in surgical training programs,"Qual. Saf. Health Care, vol. 13, pp. 19-26.

[6] F. H Garlick, "Surgical training of doctors in their own isolated hospital," Aust. N.Z. J. Surg., vol. 70, pp. 456-458, 2000.

[7] S. Grange, "A virtual university infrastructure for orthopaedic surgical training with integrated simulation," PhD, Engineering, Mathematics and Computing, University of Exeter, United Kingdom, 2006.

[8] K. Kalicki, F. Starzynski, A. Jenerowicz, K. Marasek, "Simple ossiculoplasty surgery simulation using haptic device," *mue*, pp. 932-936, 2007 International Conference on Multimedia and Ubiquitous Engineering (MUE'07), 2007.

[9] G. Welch, R. Yang, B. Cairns, H. Towles, A. State, A. Ilie, S. Becker, D. Russo, J. Funaro, D. Sonnenwald, K. Mayer-Patel, B. D. Allen, H. Yang, E. Freid, A. van Dam, and H. Fuchs, “3D Telepresence for Off-Line Surgical Training and On-Line Remote Consultation,” Proceedings of ICAT CREST Symposium on Telecommunication, Teleimmersion, and Telexistence, The University of Tokyo, Tokyo, Japan, December 2004. Invited submission.

[10] G. Welch, R. Yang, S. Becker, A. Ilie, D. Russo, J. Funaro, A. State, K.Low, A. Lastra, H. Towles, B. Cairns, H. Fuchs, and A. van Dam. “Immersive Electronic Books for Surgical Training, “ IEEE Multimedia, vol. 12, no. 3, pp. 22–35, July–September 2005.

[11] A. Ilie, k. Low, G. Welch, A. Lastra, H. Fuchs and B. Cairns, “Combining Head-Mounted and Projector-Based Displays for Surgical Training,” Presence: Teleoperators and Virtual Environments, vol. 13, no.2, pp. 128-145, April 2004.

[12] J.C. Rosser, Jr., P.J. Lynch, L. Cuddihy, D.A. Gentile, J. Klonsky, R. Merrell, “The impact of video games on training surgeons in the 21st century,” Arch. Surg., vol. 142, no. 2, pp. 181-186, Feb 2007.

[13] “The Awarepoint™ Solution”, Awarepiont: Real-time Awareness Solutions, http://awarepoint.com/Healthcare/Solution.html.

[14] “What is FLS”, Fundamentals of Laparoscopic Surgery, http://www.flsprogram.org/.

[15] “OR of the Future”, University of Maryland Medical Center, http://www.umm.edu/center/or_of_future.htm.

[16] “Zigbee”, Wikipedia, http://en.wikipedia.org/wiki/Zigbee.

[17] “Maemo”, http://maemo.org/

[18] “Nokia N800”, http://www.nseries.com/n800/.

[19] “PyBluez”, http://org.csail.mit.edu/pybluez

[20] “BlueZ’, http://www.bluez.org/

[21] “RDF-S”, http://www.w3.org/TR/rdf-schema/

[22] “ekahau”, http://ekahau.com/

[23] “Jena-A Semantic Web Framework for Java”, http://jena.sourceforge.net

[24] “ProMIS™ surgical simulator”, http://www.haptica.com/id11.htm

[25] “Human Patient Simulator”, METI Medical Education Technologies, Inc., http://www.meti.com/Product_HPS.html

J Real-Time Image Proc
DOI 10.1007/s11554-007-0028-y

ORIGINAL RESEARCH PAPER

FPGA-based real-time 3D image preprocessing for image-guided medical interventions

Omkar Dandekar · Carlos Castro-Pareja · Raj Shekhar

Received: 3 January 2007 / Accepted: 5 May 2007

Abstract Minimally invasive image-guided interventions (IGIs) are time and cost efficient, minimize unintended damage to healthy tissue, and lead to faster patient recovery. One emerging trend in IGI workflow is to use volumetric imaging modalities such as low-dose computed tomography (CT) and 3D ultrasound to provide real-time, accurate anatomical information intraoperatively. These intraoperative images, however, are often characterized by quantum (in low-dose CT) or speckle (in ultrasound) noise and must be enhanced prior to any advanced image processing. Anisotropic diffusion filtering and median filtering have been shown to be effective in enhancing and improving the visual quality of these images. However, achieving real-time performance, as required by IGIs, using software-only implementations is challenging because of the sheer size of the images and the arithmetic complexity of the filtering operations. We present a field-programmable gate array-based reconfigurable architecture for real-time preprocessing of intraoperative 3D images. The proposed architecture provides programmable kernels for 3D anisotropic diffusion filtering and 3D median filtering within the same framework. The implementation of this architecture using an Altera Stratix-II device achieved a voxel processing rate close to 200 MHz, which enables the use of these processing techniques in the IGI workflow prior to advanced operations such as segmentation, registration, and visualization.

Keywords Image-guided interventions · 3D anisotropic diffusion · 3D median filtering · Real-time image processing · Field-programmable gate array

1 Introduction

Image-guided interventions (IGIs), including surgeries, biopsies, and therapies, have the potential to improve patient care compared to conventional open and invasive procedures by enabling new and faster procedures, minimizing damage to healthy tissues, improving the effectiveness of procedures, producing fewer complications, and allowing for clinical intervention at a distance. With further invention and development of imaging and image processing techniques, it is conceivable that innovative minimally invasive IGIs will replace conventional open and invasive techniques.

Real-time and high-quality three-dimensional (3D) intraoperative visualization is a critical need for IGIs. Recent advances in computer and transducer technology have made high-speed 3D imaging possible with high resolution and acquisition speed. Notably, low-dose computed tomography (CT) and 3D ultrasound have emerged as the preferred volumetric imaging modalities during many image-guided procedures [1, 13, 18, 19, 21]. The advent of multislice CT allows high-resolution and high-frame-rate volumetric imaging of the operative field. In the continuous volumetric mode, multislice CT is capable of acquiring images with 256 × 256 × 64 dimensions and resolutions of 0.625 mm, 8 times per second. Similarly, advances in transducer technology have led to improvements in the

O. Dandekar · R. Shekhar
Department of Electrical and Computer Engineering,
University of Maryland, College Park, MD 20742, USA

O. Dandekar · C. Castro-Pareja · R. Shekhar (✉)
Department of Diagnostic Radiology and Nuclear Medicine,
N2W78, University of Maryland, 22 S. Greene Street,
Baltimore, MD 21201, USA
e-mail: rshekhar@umm.edu

field of 3D ultrasound imaging, which can now acquire images with 128 × 128 × 128 dimensions and resolution of 1 mm, 20 times per second. These intraoperative images, acquired during the procedure for navigation, represent the most current anatomical information but often suffer from poor signal-to-noise ratio. To achieve desired accuracy for IGIs, these intraoperative 3D images, therefore, must be preprocessed and enhanced before they can be used for advanced image processing operations such as segmentation, registration, and visualization. Toward this end, anisotropic diffusion filtering and median filtering have been shown to be effective in enhancing and improving the visual quality of these images. It is important to note that the interactive nature of IGIs necessitates equivalent image processing speed so that these procedures can be performed in a streamlined manner without any additional processing latency.

The aforementioned filtering techniques are based on neighborhood (window) operations. For volumetric (3D) images, these neighborhoods are considerably larger (N^3), thus increasing the complexity of filtering operations. This complexity, coupled with the sheer size of intraoperative volumetric images, results in execution times of several seconds for software implementations running on general-purpose workstations (Table 1). This processing speed is only a fraction of the acquisition speed of the intraoperative images and is clearly unacceptable to meet the real-time requirements of IGIs. Previously reported techniques for accelerated implementation of these filtering operations primarily focused on one-dimensional (1D) or two-dimensional (2D) filters [3, 17, 27, 34, 38], with only a few implementations attempting to accelerate these operations in 3D.

We present a field-programmable gate array (FPGA)-based architecture for real-time processing of intraoperative 3D images. Earlier attempts to accelerate 3D anisotropic diffusion filtering were targeted toward multiprocessor clusters [5, 35]. Despite the near-linear speedup offered by these techniques, the need to employ up to 256 processors to achieve real-time performance makes them less suitable for clinical deployment. In this work, we introduce a novel FPGA-based implementation of 3D anisotropic diffusion filtering. The reported solution is compact, easily deployable, and capable of processing the intraoperative images faster than acquisition speeds. Some researchers have recently reported high-speed implementations of 3D median filtering using graphics processing units [36] and FPGAs [23]. This manuscript presents an FPGA-based 3D median filtering module that is faster than currently existing solutions and supports higher 3D kernel sizes (3,5,7). The reported architecture can achieve a processing rate close to 200 Megavoxels per second for both the 3D anisotropic diffusion and 3D median filtering, which is equivalent to about 50 processing iterations or operations per second for images of size 256 × 256 × 64. Consequently, this design is capable of meeting the real-time data processing need of most IGIs.

2 Filtering algorithms and previous work

2.1 Anisotropic diffusion filtering

Anisotropic diffusion filtering is an iterative process which progressively smoothes an image while maintaining the significant edges. The nonlinear anisotropic diffusion algorithm for edge-preserving image smoothing was first proposed by Perona and Malik [30]. For a 3D image I with intensities $I(\bar{v},t)$, where $\bar{v}$ is a vector in the 3D space and t is a given point in time (for the purposes of modeling the diffusion process), the diffusion process is described by the following equation:

$$\frac{\partial I}{\partial t} = div(c(\bar{v},t) \cdot \nabla I(\bar{v},t)), \qquad (1)$$

where c is the diffusion coefficient and takes a value between zero and 1. In general, the diffusion coefficient is defined as a function of the image gradient [i.e., $c = f(|\nabla I|)$].

Table 1 Software execution time of 3D anisotropic diffusion filtering and 3D median filtering of 8-bit images for common kernel sizes (N)

Filter kernel	Kernel size (N)	Image size (voxels)	Execution time (s)	Voxel processing rate (MHz)
3D anisotropic diffusion filter	7	128 × 128 × 128	2.28	0.92
		256 × 256 × 64	4.58	0.92
3D median filter	3	128 × 128 × 128	0.85	2.46
		256 × 256 × 64	1.59	2.63
	5	128 × 128 × 128	3.01	0.7
		256 × 256 × 64	5.67	0.74

The last column shows the corresponding voxel processing rate in MHz. In contrast, the reported architecture is capable of a voxel processing rate close to 200 MHz

For noisy images, Whitaker and Pizer [37] showed that gradient estimates taken from the image itself tend to be unreliable and proposed, instead, the use of a Gaussian-filtered version of the image to calculate the gradient values. Their proposed Gaussian filter has a standard deviation $\sigma(t)$ that decreases as the time (t) increases, thus resulting in a multiscale approach. Dorati et al. [12] demonstrated the usefulness of Whitaker and Pizer's approach to 3D ultrasound image preprocessing. The diffusion coefficient that uses the Gaussian-filtered image [indicated as $G(\sigma(t))$] is then defined as:

$$c = f(|\nabla G(\sigma(t)) \cdot I(\bar{v}, t)|). \tag{2}$$

Several diffusion functions have been proposed in the literature. The two most widely used are:

$$c_1 = \exp\left(-\left(\frac{|\nabla G(\sigma(t)) \cdot I(\bar{v}, t)|}{K}\right)^2\right) \tag{3}$$

and

$$c_2 = \left(1 + \left(\frac{|\nabla G(\sigma(t)) \cdot I(\bar{v}, t)|}{K}\right)^{1+\alpha}\right)^{-1}. \tag{4}$$

These diffusion functions depend on the gradient of the Gaussian-filtered image, while the parameter K adjusts the levels at which edges are diffused or preserved, to achieve the desired filtering effect. The corresponding discrete expression for this filtering operation (shown for a 2D case for simplicity) is:

$$I(x,y,t+\Delta t) = I(x,y,t) + \Delta t \cdot \begin{pmatrix} c(|I_G(x+1,y,t) - I_G(x,y,t)|) \cdot [I(x+1,y,t) - I(x,y,t)] \\ +c(|I_G(x-1,y,t) - I_G(x,y,t)|) \cdot [I(x-1,y,t) - I(x,y,t)] \\ +c(|I_G(x,y+1,t) - I_G(x,y,t)|) \cdot [I(x,y+1,t) - I(x,y,t)] \\ +c(|I_G(x,y-1,t) - I_G(x,y,t)|) \cdot [I(x,y-1,t) - I(x,y,t)] \end{pmatrix},$$

where I_G is the Gaussian-filtered version of the image, c is the discrete realization of the chosen diffusion function, and the time step Δt controls the rate and stability of the diffusion process. Gerig et al. [16] calculated maximum values for Δt for different neighborhood structures. For a 3D implementation, diffusion is calculated in a 3D space with six-connected neighborhood, and that configuration corresponds to a maximum Δt value of 1/7.

Earlier efforts to accelerate 2D anisotropic diffusion filtering employed graphics hardware [34] and analog hardware [17, 38]. Accelerated implementations of 3D realizations using computing clusters have also been reported [4, 5, 35]. The work reported in this manuscript introduces a novel FPGA-based implementation of 3D anisotropic diffusion filtering. The reported implementation, using a single FPGA device, achieves a voxel processing rate better than that previously achieved by a 256-processor cluster [5]. The reported solution is compact, offers real-time performance and hence is more suitable for clinical deployment.

2.2 Median filtering

Median filtering is a nonlinear technique commonly used to eliminate speckle noise from ultrasound and impulse noise from other noisy images. Accelerated implementations of median filters based on searching, sorting, and bit-level methods have previously been reported in the literature. We particularly focus on bit-level methods because they are well suited to finding the median of large neighborhoods that commonly arise in 3D image processing. Bit-level methods for median filtering can be classified into the bit-serial sorting, bit-serial searching, threshold decomposition, and majority voting-based methods. Bit-serial sorting is performed using sorting networks such as the odd–even exchange network and reduced bubble sort network [25, 29]. Bit-serial searching [2], also called the radix method, involves a bit-by-bit search to find the median. The threshold decomposition method [14] provides a modular and parallel design, but the hardware requirements grow exponentially with the number of bits used to represent images. Majority voting methods are based on determining bit-wise majority starting from the most significant bits (MSBs). Lee and Jen [26, 27] have described a novel binary majority gate that can determine the majority of binary input signals using an inverter circuit. A compact majority voting circuit using an adder array to count the number of 1s and a threshold comparator to determine an individual bit of the median is described by Benkrid et al. [3]. Variations on this approach have been described in the literature [20, 22, 24]. Systolic array architectures for bit-level sorting networks have been shown to improve concurrency of the bit-serial sorting designs [8–10, 25, 29, 33]. The median filter design presented in this manuscript is a combination and 3D extension of bit-serial searching and majority voting approaches.

It must be noted, however, that most reported techniques for fast implementation of median filtering operation focus on 1D or 2D realizations and do not adequately address the need for accelerating this operation in 3D. More recently, a few high-speed implementations of 3D median filtering have been reported [23, 36]. The FPGA-based architecture reported by Jiang and Crookes [23] uses a partial sort algorithm. Although that work reports a voxel processing rate of around 50 MHz, the reported design cannot be extended easily to filter kernel sizes beyond 3. The GPU-based implementation reported by Viola et al. [36] is flexible in terms of kernel size but falls short of achieving real-time performance. Our work as detailed in this article presents an FPGA-based 3D median filtering module that offers a voxel processing rate close to 200 MHz. The reported design is faster than currently existing solutions and can support higher 3D kernel sizes (3,5,7).

2.2.1 *Median filtering algorithm*

As noted, the median filter design presented in this manuscript is based on a combination of bit-serial searching and majority voting approaches. This median finding scheme can be briefly described by means of an example. The algorithm is executed in b (for b-bit images) steps, where each step finds 1 bit of the resulting median value starting from the most significant to the least significant bit. Specifically, at the jth step, the majority bit ('0' or '1') among the jth significant bits of all the input elements in the neighborhood is calculated and represents the jth bit of the median of the neighborhood ($0 \le j \le b-1$). At a given step, when a bit of an input element differs from the majority bit calculated at that step, the bit value for that element is fixed in subsequent steps and is considered to be masked with its current bit value. Bits already masked in a previous step are not altered in subsequent steps (i.e., if the jth bit of input value n, represented using b bits as: n_{b-1}, n_{b-2},…,n_0, is masked, then the algorithm considers $n_i = n_j$, $\forall i<j$). The process of finding the median using this approach is illustrated in Fig. 1. In this example, for simplicity, we consider a small input neighborhood consisting of only five elements with four bits per voxel (b = 4); therefore, only four processing steps are required. Processing starts at the MSB position of the data elements. The bits of the data elements being considered for calculating the majority bit at any step are indicated in gray in the figure. The masking operation that takes place at the end of every step is indicated by arrows. The masked bits are shown to be crossed out. One bit of the median is determined at every processing stage starting from the MSB position, and the results from all the stages are combined to produce the final median value.

The FPGA-based systolic array implementation of this median-finding algorithm is described in Sect. 3.2.2. This implementation is pipelined; each step of the algorithm is mapped to a pipeline stage containing an identical processing element. Implementation for a b-bit image, therefore, requires b processing stages.

3 Architecture

We present an FPGA-based architecture that is capable of performing 3D anisotropic diffusion and 3D median filtering of intraoperative images faster than their acquisition speed. A top-level block diagram of this architecture is shown in Fig. 2. Input and output images are stored in two independent external memory banks, and the memory controller, input and output image buffers, and the filtering modules are implemented using an FPGA. The reported architecture supports two filtering modules, one for 3D anisotropic diffusion filtering and the other for 3D median filtering. The filtering module can be selected and reconfigured statically, whereas the memory controller and the image buffers are designed to be common to all supported filtering modules. The role of input and output memory banks can be switched at runtime, thus enabling execution of consecutive filtering operations (or iterations) without additional data transfers between the memory banks.

In order to achieve real-time performance, it is imperative to aim at a throughput of one processed (output) image voxel per clock cycle. Because both anisotropic diffusion and median filtering involve neighborhood operations, meeting this throughput requirement is chal-

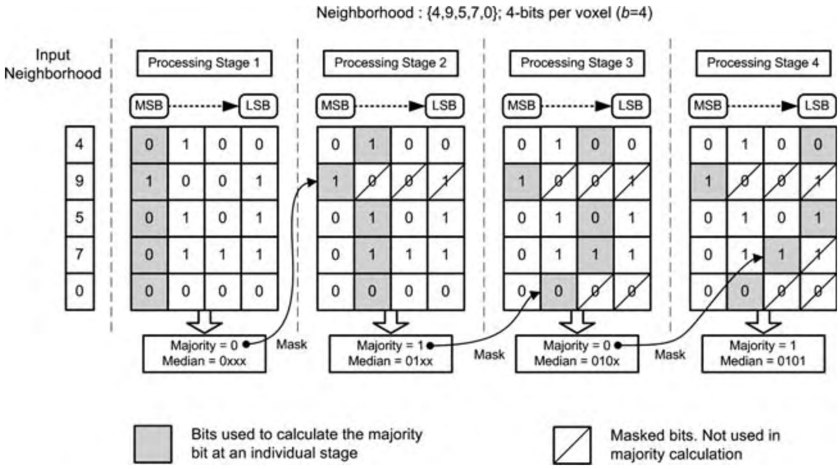


Fig. 1 A median filtering example using majority voting technique. In this example, M = 5 and b = 4. *Arrows* indicate masking operations that may happen at the end of a step

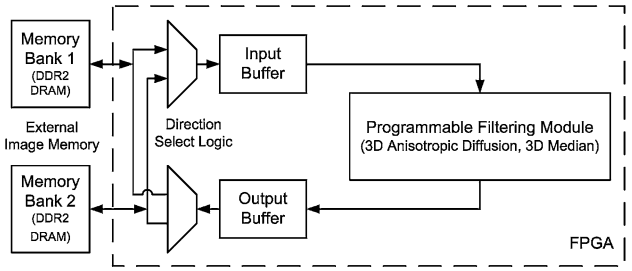


Fig. 2 Block diagram of the reported FPGA-based real-time 3D image processing system

lenging, given that an entire neighborhood (N^3 voxels, where N is the filter kernel size) must be accessed in order to compute one output voxel. Moreover, adjoining neighborhoods must be continuously fetched from the input memory bank as next output voxels are computed sequentially. These neighborhoods are read by the memory controller from the input image memory bank and are stored into the input buffer in an $N \times N \times N$ arrangement. The filtering module receives the neighborhood to be processed in a pipelined fashion: $N \times N$ new voxels every clock cycle. Once the filtering module pipeline is full (after N clock cycles), the filtering module computes one output voxel per clock cycle. The sequential input neighborhoods are continuously processed, and the resulting output voxels are stored into the output buffer. The memory controller then transfers these resultant voxels to the output image memory in a burst of one image row at a time. The memory controller uses a brick-caching scheme, specifically devised to meet the high input data rate required by this task. This brick-caching scheme takes advantage of the fact that adjacent neighborhoods share $N \times N \times (N - 1)$ voxels and only $N \times N$ new voxels need to be supplied every clock cycle for continuous neighborhood processing. The implementation of this scheme is described in Sect. 3.1. The detailed design of the supported 3D anisotropic diffusion and 3D median filtering modules is presented in Sect. 3.2.

For the remainder of this manuscript, we will use the following notations: image dimensions are represented as $N_x \times N_y \times N_z$. The parameter b indicates the number of bits used to represent the voxel intensity in the image, and N is the filter kernel size with corresponding neighborhood size of N^3. Input and output images are arranged in the memory banks along the $z - y - x$ order, with rows of the memory aligned with the z direction of the image. The output voxels are also calculated in $z - y - x$ order.

3.1 Memory controller and brick-caching scheme

Memory organization and neighborhood access techniques are often the limiting factors in 3D image processing systems [7, 11, 31, 32]. However, most practical filtering techniques employ standard neighborhood operations that require block-sequential voxel access, as shown in Fig. 3. The reported FPGA-based architecture uses a raster scan order distribution of voxels in the image memory, along with a brick-caching scheme to take advantage of this block-sequential access pattern. For every output voxel calculation, an entire neighborhood of $N \times N \times N$ voxels must be accessed. This neighborhood cannot be retrieved in a single burst access of a sequentially organized image memory. Moreover, in a pipelined implementation, data must be continuously fetched for successive neighborhood operations. To sustain the high data rate required to achieve real-time processing speeds, the reported architecture employs a brick-caching scheme that loads the image into the input buffer that stores an $N \times (N + 1)$ array of image rows (i.e., it stores up to $N \times (N + 1) \times N_z$ voxels). This input buffer is implemented using high-speed and dual-ported memory blocks internal to the FPGA. The input buffer can be accessed in a single clock cycle, which enables fast updates and reads. Figure 4 shows the block diagram of the input image memory and the input buffer, which consists of an $N \times (N + 1)$ array of internal memory blocks, each holding N_z voxel intensity values. We use the following terminology: a *brick* is an $N \times N \times N_z$ block of image

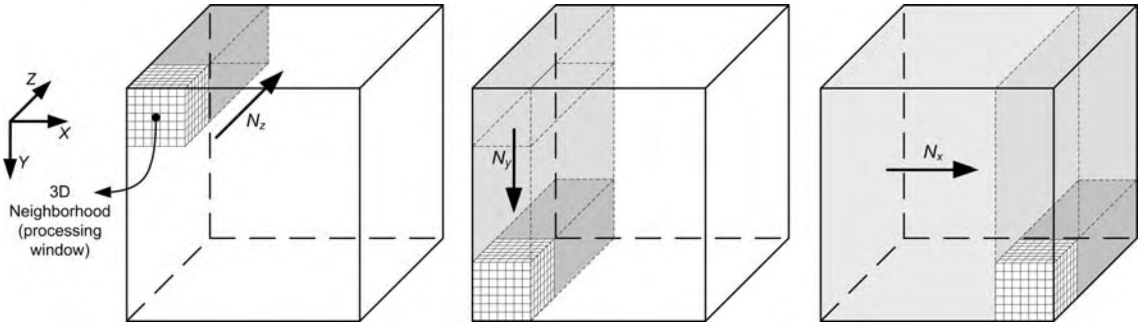


Fig. 3 Typical voxel access pattern for neighborhood operations-based image processing

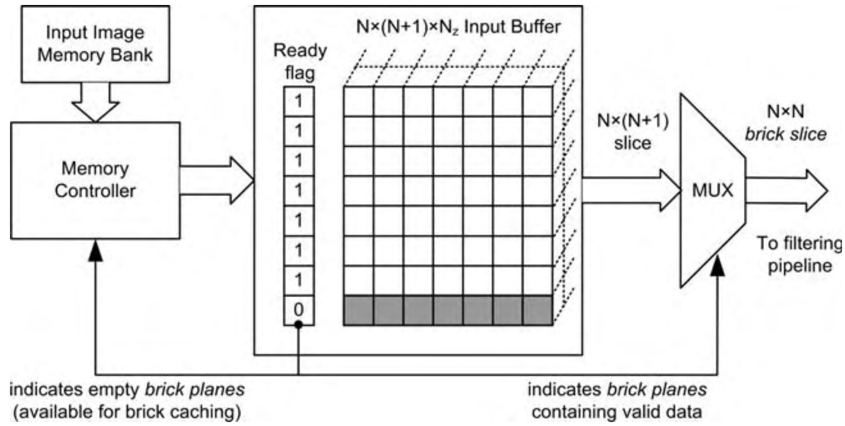

Input Image Memory Bank
Memory Controller
Ready flag
N×(N+1)×N$_z$ Input Buffer
1
1
1
1
1
1
1
0
N×(N+1) slice
MUX
N×N brick slice
To filtering pipeline
indicates empty *brick planes* (available for brick caching)
indicates *brick planes* containing valid data

Fig. 4 Block diagram showing the input image memory and the input buffer configuration

intensity values stored in the internal buffer. A *brick plane* is an $N \times 1 \times N_z$ section of a brick. A *brick slice* is an $N \times N \times 1$ section of a brick. A *brick row* (or simply row) is a $1 \times 1 \times N_z$ section of a brick. Each row corresponds to and contains one input image row containing N_z voxels. The pictorial representation of this terminology is shown in Fig. 5. Bricks are loaded into the buffer one brick row at a time for an available brick plane and are then fed to the filtering pipeline one brick slice at a time. The input buffer, therefore, can store a whole brick plus an extra brick plane. The brick-caching operation is described below.

The memory controller fills up the input buffer row by row. Each row contains N_z voxels, and thus transfer of each row takes $t_{\text{Row}} = t_{\text{lat}} + N_z/W$ clock cycles, where t_{lat} is the number of clock cycles necessary to start a burst memory transfer and W is the effective data bus width in terms of number of voxels (e.g., double-data-rate [DDR] dynamic random access memory [DRAM] will offer twice the effective bus width of single-data-rate DRAM with a similar configuration). After the first row is cached in, the controller starts caching the row next to it in the x direction. A complete brick plane (N image rows, $N \times 1 \times N_z$ voxels) can be loaded in $t_{\text{Plane}} = N \cdot t_{\text{Row}}$ clock cycles. Associated with every brick plane is a ready flag. This flag serves a dual purpose; when ‘1’, it indicates the availability of data for that particular brick plane, and, when ‘0’, it indicates that the brick plane is empty and available for caching image voxels. After one brick plane is loaded into the input buffer, the memory controller sets the corresponding ready flag and starts loading the next brick plane (along the y direction). Once a complete $N \times N \times N_z$ brick is available in the input buffer, it is fed into the filtering module pipeline one brick slice ($N \times N$ voxels) at a time.

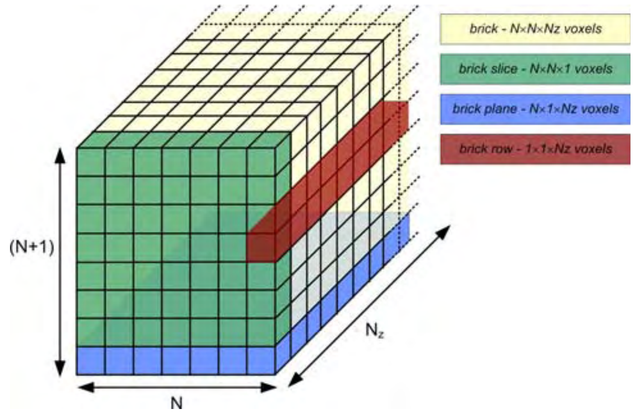

brick - N×N×Nz voxels
brick slice - N×N×1 voxels
brick plane - N×1×Nz voxels
brick row - 1×1×Nz voxels
(N+1)
N$_z$
N

Fig. 5 Pictorial representation of the notation used in the brick-caching scheme

The filtering module pipeline operates on one $N \times N \times N$ neighborhood at a time and is fed with a new brick slice every clock cycle. Loading an entire $N \times N \times N_z$ brick into the filtering module pipeline thus takes N_z clock cycles. While this operation is in progress, the memory controller loads the next brick plane (along the y direction) into the buffer plane that is not being used for processing (indicated by the ready flag), which requires t_{Plane} clock cycles. After processing of all the neighborhoods in the $N \times N \times N_z$ brick is complete, the processing window shifts along the y dimension of the image, and the processing of the new neighborhoods begins. Simultaneously, the ready flag corresponding to the brick plane that is no longer used is set to ‘0’. This available brick plane in the input buffer is then used for caching the next image rows (along y direction). In this fashion all brick planes in the input buffer are cyclically used for brick caching during processing. These steps continue until the processing window reaches the end of the column (i.e., until $y = N_y$). At this point, the processing window moves along the x direction. To accomplish this, data in the internal buffers are invalidated, and the complete $N \times N \times N_z$ brick in the next column is cached, which requires a pipeline stall. After the initial brick in the new x coordinates is loaded, the processing continues as described earlier. The processing of the entire 3D image is completed accordingly. For continuous pipelined operation with minimum stalls, the memory controller must provide the next brick plane before the processing of the previous brick is completed. Therefore, the relationship expressed in the following equation must be met:

$$N \cdot (t_{\text{lat}} + N_z/W) \cdot T_{\text{mem}} \leq N_z \cdot T_{\text{proc}}, \tag{6}$$

where T_{mem} is the clock period of the external memory clock and T_{proc} is the clock period of the internal filtering pipeline. The left-hand side of Eq. (6) refers to the total time required to load a new brick plane. The right-hand side refers to the total time required to process a whole brick. Assuming that efficient burst accesses (supported by most modern dynamic memories) are being used (which implies: $t_{\text{lat}} \ll N_z /W$), the following relationship must be maintained to minimize pipeline stalls:

$$N \cdot T_{\text{mem}} \leq W \cdot T_{\text{proc}}. \tag{7}$$

3.2 Filtering modules

3.2.1 3D anisotropic diffusion filtering

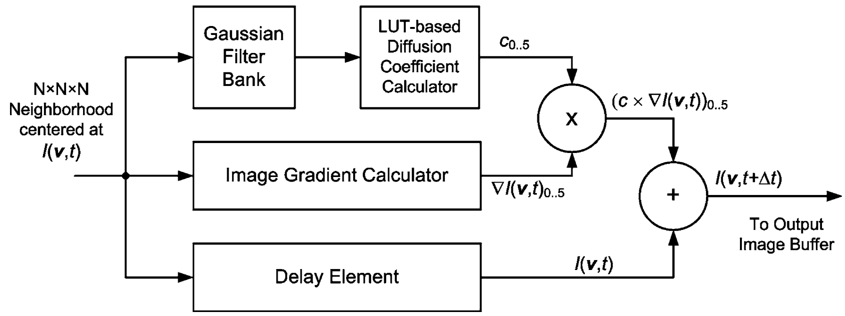
The reported FPGA-based architecture supports 3D anisotropic diffusion filtering by pipelined implementation of the 3D extension of the formulation shown in Eq. (5). As indicated by that formulation, we use a Gaussian-filtered version of the image for improved diffusion coefficient estimation. Our design implements this Gaussian filtering at runtime using an embedded 3D Gaussian filtering module. Figure 6 shows a top-level block diagram of the reported 3D anisotropic diffusion filtering module. This filtering pipeline operates on $N \times N \times N$ voxel neighborhoods. On each clock cycle, the input data buffer feeds an $N \times N$ voxel neighborhood (brick) slice into the pipeline. The center voxel intensity value is passed to the delay element for accumulation at the end of the pipeline per Eq. (5). The 3×3 voxel neighborhood located at the center of the incoming $N \times N$ neighborhood is passed to the image gradient calculator, which calculates the image gradients with respect to each of the 6-connected neighbors of the center voxel. The embedded Gaussian filtering module calculates, in parallel, the Gaussian-filtered values for each of the six-connected neighbors and passes them to the diffusion coefficient calculator, which calculates the diffusion coefficients $c_{0..5}$ corresponding to each of the input gradients. Taking advantage of the parallelism native to FPGAs, these operations are executed in parallel, and, as a result, this filtering module can calculate the output at the rate of one voxel per clock cycle. The resulting voxel intensity values are fed into the output buffer, and the memory controller then stores them in the output memory bank.

Embedded Gaussian filtering module Equation (8) shows the formula to calculate the coefficients of a 3D Gaussian filter kernel, where σ is the standard deviation of the Gaussian function and d is the Euclidean distance between the desired coefficient location and the kernel center. For a chosen σ, the coefficient values depend exclusively on the Euclidean distances from the kernel center; thus, the Gaussian filter kernel exhibits symmetries with respect to its center (i.e., it is radially symmetric).

$$G_d(\sigma) = \exp\left(-\frac{d^2}{2\sigma^2}\right) \tag{8}$$

The reported architecture takes advantage of these symmetries to reduce the number of multipliers needed for implementing the 3D Gaussian kernel to $k \times (k+1) \times (k+2)/6$ from $(N-2)^3$; where N is the size of anisotropic diffusion filtering kernel, the corresponding size of the embedded Gaussian filtering kernel is $(N-2)$, and $k = (N-1)/2$. For example, a $5 \times 5 \times 5$ embedded Gaussian kernel that arises in anisotropic diffusion filtering with $N = 7$ can be implemented using only 10 multipliers (as opposed to 125) as we reported previously [6]. For this kernel, each individual slice (5×5 plane of the kernel) has six isodistance regions and the whole 3D kernel has 10. During filtering operation, all voxels that are equidistant from the kernel center are multiplied against the same Gaussian coefficient. The intensities corresponding to these voxels in the same isodistance region can, therefore, be pre-added before being multiplied against the Gaussian

Fig. 6 Top-level block diagram of 3D anisotropic diffusion filtering. This diagram indicates paths that are executed in parallel



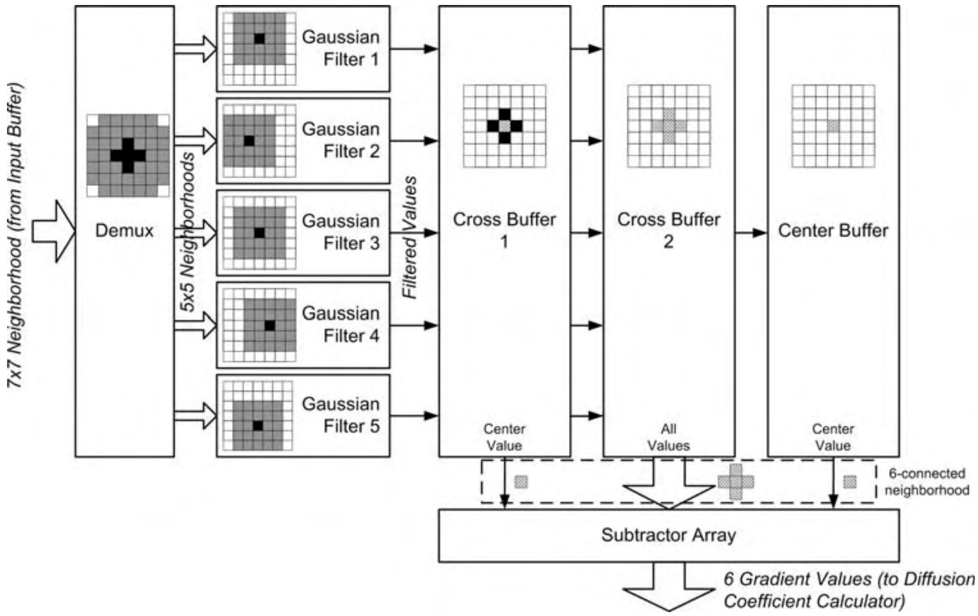


Fig. 7 Block diagram of the embedded Gaussian filter bank (for $N = 7$, corresponding Gaussian kernel size is 5)

coefficients. Because a 5 × 5 × 5 Gaussian kernel contains 10 isodistance regions, the minimum number of multipliers necessary to implement this filter kernel is, therefore, 10.

A block diagram of the Gaussian filter bank is shown in Fig. 7. On each clock cycle, the input buffer feeds an $N \times N$ voxel neighborhood into the bank. This neighborhood is decomposed into five $(N - 2) \times (N - 2)$ neighborhoods by the input demux, and these neighborhoods are then passed to five embedded 3D Gaussian filters. Figure 8 shows a block diagram of an embedded 3D Gaussian filter. The pre-adder accumulates values corresponding to the isodistance groups in the incoming $(N - 2) \times (N - 2)$ neighborhood, thus compressing the neighborhood based on the intra-neighborhood plane isodistance criterion (e.g., each single 5 × 5 slice of a 5^3 Gaussian neighborhood has six isodistance regions). These pre-added values are then passed to $(N - 2)$ pipeline buffers, which make values corresponding to the entire $(N - 2)^3$ neighborhood available in parallel. The sorter–accumulator aggregates these values corresponding to the isodistance groups between the slices, compressing them further using the isodistance criterion for the entire neighborhood (e.g., 10 values that correspond to the 10 unique coefficients in a 5 × 5 × 5 Gaussian neighborhood). These values are passed to the multiplier array, where they are multiplied against their corresponding Gaussian coefficients. The adder tree then adds the resulting values and outputs the result for the current 3D neighborhood. The results for the subsequent neighborhoods are produced continuously as a result of pipelined implementation of the operation. The Gaussian coefficients are precomputed for a given value of σ and are stored in internal registers using fixed-point representation. The effect of this fixed-point representation is analyzed in Sect. 4.1.

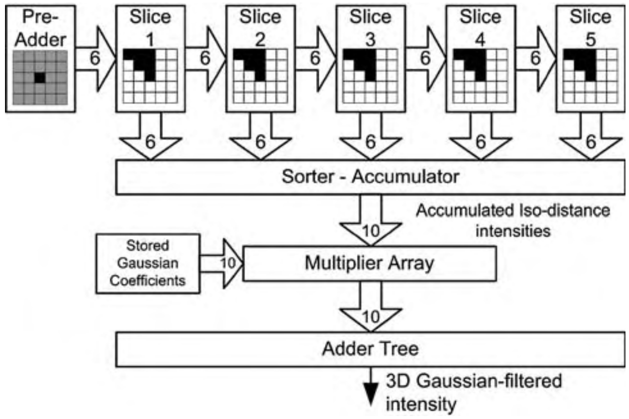


Fig. 8 Pipelined implementation of an individual Gaussian filter element (Gaussian kernel size = 5)

The results from the five individual Gaussian filters correspond to a cross-shaped region of a neighborhood slice. In order to operate on the 3D six-connected neighborhood, these results are passed to pipelined registers composed of two cross buffers and the center buffer. The cross buffers store all five values in a neighborhood slice, whereas the center buffer stores only the center value. As a result, the entire six-connected neighborhood is available between these buffers. The buffers then send the Gaussian-filtered, six-connected 3D voxel neighborhood to the subtractor array, which calculates the six corresponding gradient values and passes them to the diffusion coefficient calculator.

Diffusion coefficient calculation As noted previously, gradients calculated after Gaussian filtering are used to estimate the diffusion coefficients. For a *b*-bit image, the absolute value of the gradient is limited to the range 0 and $2^b - 1$. Taking advantage of this fact the desired diffusion function is discretized in 2^b steps and implemented using a lookup table (LUT). The use of an LUT allows an efficient implementation of any diffusion function. It must be noted that, because the dynamic range of all diffusion functions is limited to [0,1], there is no significant loss in precision by a using a fixed-point representation. The effect of this fixed-point representation is analyzed in Sect. 4.1.

Image gradient and result calculation Image gradient calculation is performed by an array of six parallel subtractors. These subtractors calculate the difference between the intensity of the voxel located in the center of the kernel against its six-connected neighbors. These values are then multiplied against their corresponding diffusion coefficients (supplied by the diffusion coefficient calculator) using an array of six parallel multipliers. The resulting filtered intensity is then obtained by adding the six results from the multipliers to the original center voxel intensity. After rounding and truncation, this result is then sent to the output buffer and is then further saved into the output memory bank.

3.2.2 Median filtering

The 3D median filtering design presented in this work is an extension of majority finding-based implementation proposed by Benkrid et al. [3]. That design was reported for a 2D realization and computed only one bit of the median value per clock cycle. All bits of the median value were obtained using a feedback loop and hence for *b*-bit images, this approach required *b* clock cycles to compute the resulting median value. The implementation reported here extends that design to 3D and unrolls the feedback loop by using multiple processing stages. Moreover, our implementation exploits the regularity of this median finding algorithm with a systolic array architecture that allows a pipelined implementation and, therefore, can achieve a throughput of one median value per clock cycle. Thus, our implementation can achieve a voxel processing speed *b* times higher than the previously reported architecture [3]. The reported linear systolic array employs *b* identical processing stages for filtering a *b*-bit image. Figure 1 illustrates execution of this algorithm for a small example and can be used to gain further insights into its hardware implementation. Each processing stage of our systolic array implementation corresponds to one step of the algorithm execution. Starting from the MSB, each stage generates one bit of the resulting median value of the neighborhood being processed. We first describe the operation of an individual processing element and then explain the functioning of the entire linear systolic pipeline, which contains *b*-processing elements.

Processing element The processing element is the atomic unit of the proposed linear systolic array design. A functional block diagram of the processing element at the *j*th stage is shown in Fig. 9. The data inputs to this processing element are Data_Bits_j and Next_Data_Bits_j, the N^3 bits used considered for majority calculation and the (*j*+1)th significant bits (from MSB) of the N^3 neighborhood elements, respectively. It must be noted that, although Next_Data_Bits_j are corresponding image intensity bits, Data_Bits_j are provided by the (*j*–1)th processing stage and may have been masked in the earlier stages. The accompanying input Mask_Bits_j is a binary flag that indicates the bits in Data_Bits_j that have been masked in the prior stages. A processing element performs two important tasks. First, it computes the majority bit within the N^3 input data bits (Data_Bits_j); second, it performs the masking operation based on the majority bit calculated and outputs masked data bits ($\text{Data\_Bits}_j + 1$) and the corresponding binary flag (Mask_Bits_{j+1}) to be used in the next processing stage. The units that perform these two operations are described below.

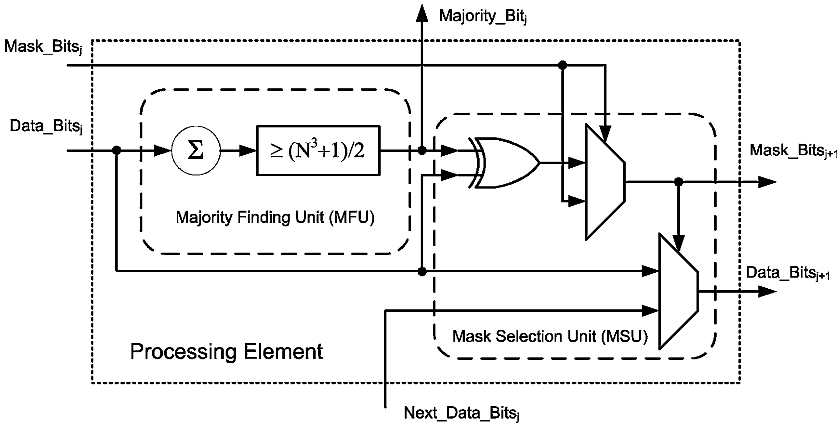


Fig. 9 A single stage (processing element) of the linear systolic median filtering kernel

Majority finding unit (MFU): The MFU consists of a bit-counting circuit that counts the number of 1s in the input bits that are considered for majority calculation (Data_Bits_j). This counting is performed using a bit adder tree customized for a chosen neighborhood size. This count is then compared against a threshold, which is programmed to be half of the number of elements contained in the neighborhood. The binary result of this comparison is the *j*th significant (from MSB) bit of the output median value. The highly compact, pipelined, and customized implementation of the MFU minimizes the combinational delay within the processing element.

Mask selection unit (MSU): After the median bit has been calculated, the MSU performs the masking operation. It computes the mask bit for each bit of Data_Bits_j, based on whether it matches with the majority bit or not. In addition, it considers and preserves the bits that were masked in the prior stages (Mask_Bits_j). Thus, the mask calculated at the jth stage (Mask_Bits_{j+1}) is a combination of the mask bits from the prior stages and the mask calculated at the current stage. This masking operation is implemented using an exclusive OR (XOR) operation and two multiplexing operations for each data bit. The calculated mask (Mask_Bits_{j+1}) is then used to selectively generate Data_Bits_{j+1} from the input Next_Data_Bits_j, while ensuring that values corresponding to the masked bits are preserved. Data_Bits_{j+1} is then used in the next processing stage to calculate the (j + 1)th significant bit of the median value.

Linear systolic design for median finding The proposed linear systolic design is realized by cascading *b* processing elements for filtering *b*-bit images. On every clock cycle, a complete neighborhood containing N^3 voxels, *b*-bits each, is fed to this linear systolic array. However, the processing stage (j + 1) can not perform its operation until stage *j* finishes its processing and provides Data_Bits_{j+1} and Mask_Bits_{j+1}. Similarly, stage *j* produces its output (*j*th significant bit of the median) one clock cycle earlier than the corresponding output by the stage (j + 1). In order to compensate for these delay and processing latencies and to provide synchronized operation, additional line delay units and data shift registers must be inserted at the input and output of the systolic array. Figure 10 shows a diagram of this configuration with *b* processing elements and required delay buffers. These delays are introduced for synchronization only, and it must be noted that as long as the input sequential neighborhoods are continuously supplied (new N^3 voxels every clock cycle), the reported systolic array design is capable of computing one median result per clock cycle. This result is then sent to the output buffer and subsequently saved into the output memory bank by the memory controller.

For correct operation, the Mask_Bits_1 input of the first processing stage (stage 1, MSB) is grounded (set to ''0''), indicating that no bits from the input data (Data_Bits_1) are masked. Also, in the final processing stage (stage *b*, LSB), Mask_Bits_{b+1} and Data_Bits_{b+1} do not need to be calculated, because the next stage does not exist. Consequently, the MSU is not needed in the final stage, which contains only an MFU to compute the last median bit. In general, for large 3D neighborhoods, the speed of the MFU is the limiting factor of the systolic array performance. In applications requiring high voxel throughput and large filtering kernels, the operation of the MFU can be pipelined. However, in those cases the depth of the delay elements used to synchronize the inputs and outputs of the different processing elements must be adjusted.

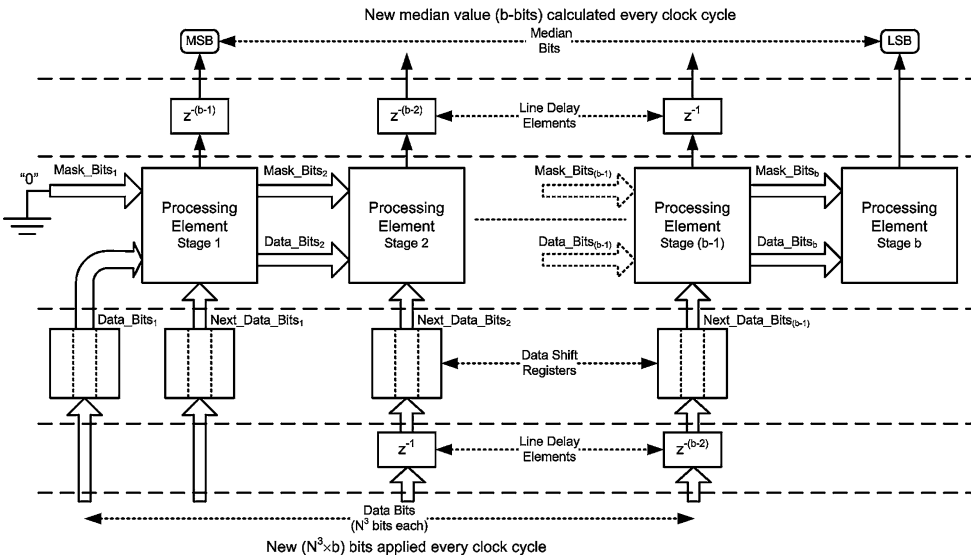

New median value (b-bits) calculated every clock cycle
MSB
Median Bits
LSB
$z^{-(b-1)}$
$z^{-(b-2)}$
Line Delay Elements
z^{-1}
"0"
Mask_Bits_1
Processing Element Stage 1
Mask_Bits_2
Data_Bits_2
Processing Element Stage 2
$\text{Mask\_Bits}_{(b-1)}$
$\text{Data\_Bits}_{(b-1)}$
Processing Element Stage (b-1)
Mask_Bits_b
Data_Bits_b
Processing Element Stage b
Data_Bits_1
Next_Data_Bits_1
Next_Data_Bits_2
$\text{Next\_Data\_Bits}_{(b-1)}$
Data Shift Registers
z^{-1}
Line Delay Elements
$z^{-(b-2)}$
Data Bits (N^3 bits each)
New (N^3×b) bits applied every clock cycle

Fig. 10 Linear systolic array architecture for median filter kernel using majority voting technique

3.3 Stall management

Pipelined implementation of a task is known to improve throughput by allowing concurrent execution of the subtasks within that task. This performance gain, however, is limited if the pipeline must be halted either because of unavailability of input data or because of interdependence between the input data. In this section, we identify the important sources of pipeline stalls in the reported architecture, describe how these stalls are managed, and analyze the effect of these stalls on the voxel processing rate of the architecture.

The architecture described in this article performs 3D filtering over an input image using neighborhood operations. As explained earlier, this architecture takes advantage of sequential neighborhood processing and employs a brick-caching scheme to achieve a high voxel processing rate. The median and the anisotropic diffusion filtering modules are designed to have no internal pipeline stalls and to produce one output voxel per clock cycle as long as input sequential neighborhoods (i.e., $N \times N$ new voxels every clock cycle) are continuously fed. The memory controller fetches the neighborhoods to be processed from the input memory bank and is responsible for maintaining this continuous flow of sequential neighborhoods. Thus, using the analogy from the producer–consumer model, the memory controller acts as a *producer*, whereas the filtering module acts as a *consumer*. The input buffer, which is an integral component of the aforementioned brick-caching scheme, acts as a buffer between the producer and the consumer. The *ready flags* associated with the brick planes of the input buffer control the status of the pipeline. When N out of the $N + 1$ brick planes available in the input buffer are valid (i.e., have ready flag = ‘1’), which implies that a complete brick is available for processing, the consumer pipeline is enabled; otherwise, it is disabled. This enabling and disabling of the pipeline is achieved by using the clock-enabling feature offered by modern FPGAs. When the pipeline is enabled, the same ready flag is also used to select and output the valid N brick planes, which are then processed by the filtering module (consumer). At the producer end, the ready flag with status ‘0’ indicates a brick plane that is empty and, hence, available for caching the image voxels. The producer continues to fetch the voxels from the input memory until the input buffer is full (i.e., all brick planes show ready flag = ‘1’). The interplay between the producer and consumer rates and the size of the intermediate buffer dictates the number of stalls incurred by the pipeline. Equation (7) summarizes this relation and establishes the criterion to achieve minimum pipeline stalls. Modern memories are becoming faster, denser, and wider, and most memory configurations can meet this criterion easily. For example, assuming that $T_{\mathrm{mem}} = T_{\mathrm{proc}}$, for a 64-bit wide DDR memory containing 8-bit images ($b = 8$, $W = 16$), kernel sizes (N) up to 16 can be supported while satisfying Eq. (7), which is sufficient for most practical filtering operations.

Once Eq. (7) is satisfied, the stalls in the processing pipeline occur only when the brick-caching scheme changes the columns (i.e., every time $y = N_y$ and $z = N_z$), while processing neighborhoods sequentially. As described earlier, at this point the pipeline must be stalled and an entire brick ($N \times N \times N_z$ voxels) from the next column must be fetched. Every time the column is changed, all the brick planes in the input buffer are invalidated (ready flag = ‘0’), which halts the consumer pipeline and prompts the producer pipeline to fetch the next brick. This happens N_x times during processing of an entire image ($N_x \times N_y \times N_z$ voxels). The stall time introduced by this mechanism is:

$$\begin{aligned}\text{Total stall time} &= \text{Number of stalls} \times \text{Time per stall}\\ &= N_x \times (N \cdot N \cdot (t_{\mathrm{lat}} + N_z/W) \cdot T_{\mathrm{mem}})\\ &\approx N_x \cdot N^2 \cdot N_z \cdot T_{\mathrm{mem}}/W \ldots \text{as}\, t_{\mathrm{lat}} \ll N_z/W,\end{aligned} \tag{9}$$

where time per stall is the time required to fetch a new brick as described in Sect. 3.1. The system processing overhead incurred by these stalls is expressed as:

$$\begin{aligned}&\text{Processing overhead}\\ &\quad = \text{Total stall time/Total processing time}\\ &\quad = (N_x \cdot N^2 \cdot N_z \cdot T_{\mathrm{mem}}/W)/(N_x \cdot N_y \cdot N_z \cdot T_{\mathrm{proc}})\\ &\quad = (N^2 \cdot T_{\mathrm{mem}})/(N_y \cdot W \cdot T_{\mathrm{proc}}).\end{aligned} \tag{10}$$

This processing overhead will reduce the effective voxel processing rate of the system from the theoretical processing rate of $1/T_{\mathrm{proc}}$; however, for typical dimensions of intraoperative 3D images and common kernel sizes this performance drop is only a small fraction of the theoretical processing rate. For example, an implementation with kernel size (N) of 5, an 8-bit image with y-dimension (N_y) equal to 256 and a 64-bit wide DDR image memory ($W = 16$) will have an overhead of only 0.6 % (assuming $T_{\mathrm{mem}} = T_{\mathrm{proc}}$).

4 Implementation and results

The reported architecture was implemented using an Altera Stratix II EP2S180F1508C4 FPGA (Altera Corp., San Jose, CA) with two external memory banks to serve as input and output image memory. The memory banks used were 1-GB DDR2 small-outline dual-inline memory DRAM modules with 64-bit data bus (i.e., $W = 16$, for $b = 8$) running at a 200 MHz clock speed. The architecture was designed using

VHSIC hardware description language (VHDL) and synthesized using Altera Quartus II 6.1. The memory controller was also implemented using VHDL and was built around the DDR2 DRAM controller megacore supplied with Altera Quartus II. Both filtering modules were custom designed using VHDL, as per the design description in Sect. 3.2. Functional verification and postsynthesis timing simulation for the entire system were performed using Modelsim SE 6.2 (Mentor Graphics, San Jose, CA). For this purpose, DDR2 DRAM simulation models provided by Micron (http://www.micron.com) were used. The reported design was then synthesized to support 8-bit images ($b = 8$) and, consequently, all results in this section are presented for 8-bit images. The execution speed of the reported architecture was obtained from postsynthesis timing simulation of the design.

4.1 Effects of fixed-point representation

Real-time filtering performance offered by the reported design is critical for the time-sensitive nature of IGIs, but of equal importance is the accuracy of the filtering process. Most software implementations represent the arithmetic operations involved in the filtering algorithms using a double precision floating-point format. This format offers high dynamic range and precision, which may or may not be required depending on the filtering technique to be implemented. Median filtering, for example, is performed exclusively using integer data (because digital images are represented using b-bit integer data), and, hence, there is no loss in precision by using a fixed-point implementation with sufficient dynamic range (i.e., using b bits for b-bit images). Our implementation of the 3D median filtering uses b-bit integer representation for b-bit images and therefore provides identical results to those provided by a software implementation.

Anisotropic diffusion filtering, however, involves operations with the data in real format. Our implementation, for the sake of efficiency in area and execution speed, used fixed-point representation to implement these arithmetic operations. We analyzed the effect of the number of bits used for this fixed-point representation on the filtering accuracy, by treating a software (C++) implementation employing double-precision floating-point representation as a reference. This analysis was performed with an 8-bit image with dimensions 256 × 256 × 64. There are two sources at which error resulting from fixed-point precision can affect the accuracy of the filtering operation: embedded Gaussian filtering and diffusion function calculation. To gain additional insight, we evaluated accuracy for these individual sources and the accuracy of the anisotropic diffusion filtering as their combined effect.

Table 2 Average error in intensity per voxel for a Gaussian filtered (kernel size, $N = 5$) 8-bit image resulting from fixed-point representation of Gaussian coefficients employed in the reported architecture

σ of the Gaussian kernel	Average error in intensity per voxel resulting from fixed-point representation of Gaussian kernel coefficients		
	8-bits	12-bits	16-bits
0.3	0.20 ± 0.47	0.07 ± 0.26	0.004 ± 0.06
0.5	0.63 ± 0.68	0.02 ± 0.11	0.004 ± 0.07
0.7	0.42 ± 0.81	0.03 ± 0.16	0.003 ± 0.06
1.0	0.21 ± 0.47	0.001 ± 0.04	0.001 ± 0.03

The error (mean ± standard deviation) is reported for common choices of σ (for Gaussian kernel) and the number of bits used to represent the Gaussian coefficients

Table 2 presents the average error in intensity per voxel after embedded Gaussian filtering (kernel size, $N = 5$) where the Gaussian kernel coefficients are represented using the fixed-point format with the designated number of bits. Because the Gaussian kernel was normalized, all coefficients were within the range [0,1], and, hence, we used one bit to represent the integer part and the rest for the fractional part. We performed this analysis for typical choices of σ for a Gaussian kernel size of 5, which corresponds to the anisotropic diffusion filtering kernel size of 7. The average error for various choices of σ with 8-bit representation is less than one intensity value, and, as expected, the average error reduces with the increasing number of bits. It must be noted, however, that embedded Gaussian filtering is used only to estimate the diffusion coefficients, and, hence, small errors introduced in this operation may not have a significant impact on the final anisotropic diffusion filtered intensity value. Because this design supported 8-bit images, a 256-entry LUT was used to implement the diffusion function shown in Eq. (3). We implemented this function for reasonable choices of the parameter K, which controls the level of the gradient at which edges are diffused or preserved. The value of K depends on the image modality and the amount of edge preservation desired. For ultrasound and low-dose CT images, however, its value is typically less than 20% of the intensity range. As the selected diffusion function takes values in the range [0,1], we used one bit to represent the integer part and the rest for the fractional part. Table 3 presents the average error per sample of diffusion function resulting from fixed-point representation with the designated number of bits for various choices of K. Although the average error increases with the choice of K, its mean and standard deviations are consistently less than 0.2% of the data range, even with representation using 8-bits. Finally, Table 4 reports average error in intensity per voxel resulting from the combined effect of finite precision

Table 3 Average error per sample of diffusion function resulting from fixed-point representation of diffusion coefficients employed in the reported architecture

K	Average error per sample of diffusion function resulting from fixed-point representation of diffusion coefficients		
	8-bits	12-bits	16-bits
10	$42 \times 10^{-5} \pm 97 \times 10^{-5}$	$3 \times 10^{-5} \pm 7 \times 10^{-5}$	$<10^{-5}$
20	$78 \times 10^{-5} \pm 114 \times 10^{-5}$	$6 \times 10^{-5} \pm 8 \times 10^{-5}$	$<10^{-5}$
30	$121 \times 10^{-5} \pm 125 \times 10^{-5}$	$9 \times 10^{-5} \pm 8 \times 10^{-5}$	$<10^{-5}$
50	$196 \times 10^{-5} \pm 116 \times 10^{-5}$	$13 \times 10^{-5} \pm 7 \times 10^{-5}$	$<10^{-5}$

The diffusion function implemented is as shown in Eq. (3), and K is the parameter that adjusts the levels at which edges are diffused or preserved. The error (mean ± standard deviation) is reported for typical choices of K and the number of bits used to represent the diffusion coefficients

Table 4 Average error in intensity per voxel for anisotropic diffusion filtered 8-bit image (kernel size, $N = 7$; Gaussian filtering with $\sigma = 0.5$; and diffusion function as shown in Eq. (3) with $K = 20$) resulting from fixed-point representation of Gaussian coefficients and the diffusion function employed in the reported architecture

Number of filtering iterations	Average error in intensity per voxel resulting from fixed-point representation of the Gaussian kernel coefficients and diffusion coefficients		
	8-bits	12-bits	16-bits
1	0.008 ± 0.092	0.001 ± 0.018	<0.001
3	0.021 ± 0.144	0.001 ± 0.031	<0.001
5	0.030 ± 0.171	0.002 ± 0.039	<0.001

The error (mean ± standard deviation) is reported for the number of bits used to represent Gaussian coefficients and the diffusion function after multiple iterations of the filtering operation

implementation of both the Gaussian coefficients and the diffusion function. For this analysis, we used the same number of bits for fixed-point representation of both entities, with one bit for the integer part and the rest for the fractional component. The kernel size (N) of the anisotropic diffusion filter was chosen to be 7, with embedded Gaussian filtering with $\sigma = 0.5$, and the diffusion function shown in Eq. (3) was implemented with $K = 20$. To evaluate error accumulation over multiple iterations of anisotropic diffusion filtering, we performed this analysis up to five iterations, which is typical for filtering of intraoperative images. The average error in intensity increases with the number of iterations, but its mean and standard deviations are less than 0.07% of the intensity range after five iterations, even with 8-bit representation.

Overall, our precision analysis indicates that even when using 8-bit fixed-point representation to perform Gaussian filtering and diffusion function calculation, the average error in intensity is only a very small percentage of the intensity range. Such small errors in intensity may not be significant for advanced operations such as registration, segmentation, and visualization and are unlikely to affect the accuracy and precision of IGIs. Our implementation, therefore, uses 8-bit fixed-point representation for these operations.

4.2 Hardware requirements

Table 5 lists the significant hardware requirements for the important modules in the proposed architecture, parameterized on filter kernel size (N) and the number of bits used

Table 5 Hardware requirements for the proposed architecture parameterized on the filter kernel size (N), number of bits per image voxel (b), and image dimension in the z direction (N_z), k is a derived parameter and is equal to $(N - 1)/2$

Hardware module	Significant hardware resources		Logic resources and performance (as implemented)		
	Multipliers ($b \times b$ bit)	Internal memory (bits)	N	Number of ALUTs (% utilization)	$f_{\max}$ (MHz)
Input buffer and controller	–	$(N \times (N + 1) \times N_z) \times b$	7	1,957 (1.5%)	233
Output buffer and controller	–	$(2 \times N_z) \times b$	7	1,743 (1.5%)	233
Anisotropic diffusion filter	$5 \times \frac{k \times (k+1) \times (k+2)}{6}$	$(3 \times 2^b) \times b$ (diffusion coefficient storage)	7	3,824 (3%)	236
Median filter	–	–	5	11,308 (8%)	224

The logic resources are reported in terms of the number of adaptive lookup tables (ALUTs) required. The percentage utilization of the logic resources is reported with respect to the target device (Altera Stratix II EP2S180F1508C4). Logic resources and the maximum operating frequency that can be achieved ($f_{\max}$) are reported for 8-bit images ($b = 8$) and a typical value of N as indicated in the fourth column to represent the voxel intensity (b). The parameter k, introduced in the context of 3D anisotropic diffusion filtering, represents the number of unique isodistances in the Gaussian kernel and is related to the filter kernel size N (usually odd) as:

$$k = \frac{(N-1)}{2}. \qquad (11)$$

The linear systolic array implementation of the 3D median filter requires logic resources only, and the resource requirements for important components of this filter kernel are listed separately in Table 6. These two tables indicate how the hardware requirements of our architecture scale with the parameters N and b. As dictated by the resource limitations imposed by the target device (Altera Stratix II EP2S180F1508C4) and real-time speed requirements, our current implementation can support filter kernel sizes (N) from the list {5,7} and {3,5} for anisotropic diffusion filtering and median filtering, respectively. The corresponding kernel sizes for the embedded Gaussian filtering in the case of anisotropic diffusion filter supported by our architecture are {3,5}. Table 5 also lists the absolute and percentage logic resources consumed by the important modules in the architecture and the maximum operating frequency ($f_{\max}$) at which these modules can run for a specific instantiation (choice of N). The percentage logic resources are reported in reference to the target device Altera Stratix II EP2S180F1508C4. The images used for this performance and logic consumption analysis were 8-bit images (b = 8). The choices for the value of N for this analysis represent common kernel size choices and are listed in the fourth column of the table.

4.3 Filtering performance

The 3D median filtering module and 3D anisotropic diffusion filtering module were synthesized for kernel sizes of {3,5} and 7, respectively, for filtering 8-bit images. The rest of the system, including the memory controller and input and output buffers, was parametrically synthesized to support the desired filtering operation and kernel size. The entire system was clocked at 200 MHz, which also corresponds to the filtering pipeline frequency (i.e., T_{proc} = 5 ns). The image memories were also clocked at 200 MHz (T_{mem} = 5 ns). For this configuration, Table 7 reports the execution time for 3D anisotropic diffusion filtering and 3D median filtering as obtained during post-synthesis timing simulation of the entire system. The image sizes used for this measurement correspond to typical dimensions of intraoperative images.

Table 6 Hardware requirements for the components of the reported linear systolic implementation of the 3D median filtering. The requirements are parameterized on the filter kernel size (N) and the number of bits per image voxel (b)

Hardware component	Number required
Processing elements	b
Data registers	$b \times N^3$
Mask select registers	$b \times N^3$
Data pipeline registers	$(b-1) \times N^3$
Line delay elements	$\frac{(b-2)\times(b-1)}{2} \times (N^3+1)$

As indicated in Table 7, the reported implementation of 3D anisotropic diffusion filtering and 3D median filtering can easily achieve a processing rate of 46 frames per second (fps) for images of size 256 × 256 × 64 voxels, which is a typical size of an intraoperative volumetric CT scan. The corresponding processing rate for an intraoperative 3D ultrasound scan with typical dimensions of 128 × 128 × 128 voxels is around 92 fps. For iterative operations such as anisotropic diffusion filtering or sequential filtering operations, this processing rate translates to 18 fps with five iterations (or sequential operations) per frame, which is sufficient to meet the real-time needs of most IGIs.

5 Discussion

We have presented an FPGA-based architecture aimed at real-time processing of intraoperative images during IGIs. The reported architecture supports 3D median filtering and 3D anisotropic diffusion filtering, which are commonly employed to enhance the visual quality of intraoperative images prior to advanced image processing techniques such as segmentation, registration, and visualization. Previously reported techniques for accelerated implementation of these filtering operations have focused for the most part on 1D or 2D realizations [3, 17, 26, 34, 38], with only a few studies addressing the need for real-time processing of volumetric images [5, 15, 23, 35, 36]. We presented an implementation of 3D median filter that is faster than existing solutions and, unlike a previously reported FPGA-based implementation [23], can be extended to kernel sizes beyond three. We also reported a novel FPGA-based implementation of 3D anisotropic diffusion filtering. The voxel processing rate achieved by the reported architecture is around two orders of magnitude higher than corresponding software implementation and compares favorably with those reported by earlier high-speed implementations.

Tables 8 and 9 compare the execution speed of the reported architecture for 3D anisotropic diffusion filtering and 3D median filtering, respectively, against a corresponding software implementation and previously reported high-speed implementations using different computing platforms. The execution time has been normalized by the

Table 7 Execution time of 3D anisotropic diffusion filtering and 3D median filtering of 8-bit images with common kernel sizes (*N*) using the reported architecture

Filter kernel	Kernel size (N)	Image size (voxels)	Execution time (milliseconds)	Voxel processing rate (MHz)
3D anisotropic diffusion filter	7	128 × 128 × 128	10.90	192
		256 × 256 × 64	21.63	194
3D median filter	3	128 × 128 × 128	10.75	195
		256 × 256 × 64	21.44	196
	5	128 × 128 × 128	10.82	194
		256 × 256 × 64	21.58	194

image dimensions for all implementations, and the performance is presented in terms of voxel processing rate to facilitate a fair comparison independent of image dimensions. The software implementation was developed using C++, and its performance was measured on an Intel Xeon 3.6 GHz workstation with 2 gigabytes of DDR2 400 MHz main memory. Although the reported architecture can support various kernel sizes for the filtering operations, for consistency the performance has been compared for a kernel size (*N*) common to all implementations: $N = 3$ for the median filtering and $N = 7$ for anisotropic diffusion filtering.

As indicated by Table 8, the reported implementation of 3D anisotropic diffusion filtering provides more than two orders of magnitude speedup over the software implementation using a single workstation. Moreover, the performance of the current architecture represents an improvement over a corresponding implementation using a 256-processor computing cluster reported previously [5]. Our work presented a novel FPGA-based implementation of 3D anisotropic diffusion filtering. Salient features of this filtering module are an embedded Gaussian filtering implementation that minimizes the number of multipliers and a pipelined design that allows throughput of one output voxel per clock cycle. This filtering module offers the flexibility to support several anisotropic diffusion techniques previously reported in the literature. For example, the multiscale approach proposed by Whitaker and Pizer [37] can be implemented by changing the embedded Gaussian filter coefficients at the end of each iteration, and a time-dependent diffusion function [28] can be implemented by reprogramming the values in the diffusion function LUT. One limitation of this filtering module is the limit on the size of the embedded Gaussian filter kernel; implementing Gaussian kernels larger than seven would result in prohibitively high hardware requirements. Such large kernels, however, are uncommon in most applications. Although the reported architecture performs some of the arithmetic functions using fixed-point representation, the variation in the output intensity values is only a small fraction of the intensity range, and these variations are unlikely to affect the accuracy and precision of IGIs.

Table 9 compares the performance of the FPGA-based 3D median filtering operation described in the current work with previously reported high-speed implementations. The present implementation provides more than an order of magnitude speedup over software- and GPU-based 3D implementations and DSP-based 2D implementation. Jiang and Crookes [23] recently reported an FPGA-based 3D implementation that is capable of achieving a voxel processing rate of 50 MHz. That design, however, was based on a partial sorting technique and cannot be easily extended to kernel sizes beyond 3. Our reported implementation, in contrast, achieved a superior voxel processing rate and is sufficiently compact to allow implementation of kernel sizes up to seven, which is sufficient for most common image processing tasks. The logic resources required by the described systolic array-based median filter indeed scale up as kernel sizes get larger; but as modern FPGAs become more dense and offer improved logic capacity, this

Table 8 Performance comparison of the reported FPGA-based 3D anisotropic diffusion filtering with a software implementation and previously reported high-speed implementations

Implementation	Platform	Filter kernel dimensionality	Voxel processing rate (MHz)	Speedup offered by the reported architecture
Software (C++)	Xeon workstation	3D	0.92	208.00
Bruhn et al. [5]	256-processor cluster	3D	105.00	1.83
Tabik et al. [35]	16-processor cluster	3D	5.66	33.92
Reported architecture	FPGA	3D	192.00	–

The last column shows the speedup offered by the reported implementation

Table 9 Performance comparison of the reported FPGA-based 3D median filtering with a software implementation and previously reported high-speed implementations

Implementation	Platform	Filter kernel dimensionality	Voxel processing rate (MHz)	Speedup offered by the reported architecture
Software (C++)	Xeon workstation	3D	2.63	74.14
Viola et al. [36]	GPU	3D	0.76	256.58
Gallegos-Funes and Ponomaryov [15]	DSP	2D	4.50	43.33
Jiang and Crookes [23]	FPGA	3D	50.00	3.90
Reported architecture	FPGA	3D	195.00	–

The last column shows the speedup offered by the reported implementation

requirement is still a small percentage of the total available resources (see Table 5). Also, for larger kernel sizes, the maximum operating frequency of the median filtering module primarily depends on the time required to find the majority bit within a processing element of the systolic array. However, further pipelining of the MFU allows compensation for the added complexity resulting from larger neighborhoods and thus improves the performance. The reported design employs this technique to achieve voxel processing rates close to 200 MHz for kernel sizes up to 5 (see Table 7).

Notwithstanding its limitations, this work presents a novel FPGA-based architecture for real-time preprocessing of intraoperative volumetric images. The reported architecture is capable of image processing rates faster than intraoperative image acquisition speeds and therefore can meet the data processing needs of most IGIs. The real-time performance and accuracy offered by the reported architecture in conjunction with its compact implementation makes it ideally suited for clinical deployment.

6 Conclusion

We have presented an FPGA-based architecture for real-time preprocessing of volumetric images acquired during IGIs. The reported architecture enables 3D anisotropic diffusion filtering and 3D median filtering of intraoperative images at the rate of 90 fps, which is faster than current acquisition speeds. The solution presented offers real-time performance, is compact and accurate, and, hence, suitable for integration into IGI workflow.

Minimally invasive IGIs are efficient, lead to faster recovery, and as a result are becoming increasingly popular. Fast and high-quality volumetric imaging and subsequent visualization are critical for the success of these procedures. Intraoperative imaging modalities continue to offer wider coverage and higher imaging speed, with a corresponding need for real-time processing of these images. The real-time performance of the reported design along with the throughput of one voxel per cycle can respond to these 4D (3D + time) image processing needs.

Acknowledgments This work was partially supported by the Department of Defense grant DAMD17-03-2-0001. The authors would like to thank Dr. Vivek Walimbe and Dr. Nancy Knight for their help in editing and refining this manuscript. The authors also thank the anonymous reviewers for their feedback and suggestions in improving this manuscript.

References

1. Antoch, G., Debatin, J.F., Stattaus, J., Kuehl, H., Vogt, F.M.: Value of CT volume imaging for optimal placement of radiofrequency ablation probes in liver lesions. J. Vasc. Interv. Radiol. **13**(11), 1155 (2002)
2. Ataman, E., Alparslan, E.: Applications of median filtering algorithm to images. Electronics Division, Marmara Research Institute, Gebze, Turkey (1978)
3. Benkrid, K., Crookes, D., Benkrid, A.: Design and implementation of a novel algorithm for general purpose median filtering on FPGAs. In: Proceedings of the IEEE International Symposium on Circuits and Systems, ISCAS, vol. 4, pp. 425–428 (2002)
4. Bruhn, A., Jakob, T., Fischer, M., et al.: Designing 3D nonlinear diffusion filters for high performance cluster computing. In: Proceedings of the 24th DAGM Symposium on Pattern Recognition, vol. 2449, pp. 290–297 (2002)
5. Bruhn, A., Jakob, T., Fischer, M., et al.: High performance cluster computing with 3D nonlinear diffusion filters. Real Time Imaging **10**(1), 41–51 (2004)
6. Castro-Pareja, C.R., Dandekar, O.S., Shekhar, R.: FPGA-based real-time anisotropic diffusion filtering of 3D ultrasound images. Proc. Real Time Imaging IX SPIE **5671**, 123 (2005)
7. Castro-Pareja, C.R., Jagadeesh, J.M., Shekhar, R.: FAIR: a hardware architecture for real-time 3D image registration. IEEE Trans. Inf. Technol. Biomed. **7**(4), 426–434 (2003)
8. Chakrabarti, C.: High sample rate array architectures for median filters. IEEE Trans. Signal Process. **42**(3), 707–712 (1994)
9. Chang, L.W., Lin, J.H.: A bit-level systolic array for median filter. IEEE Trans. Signal Process. **40**(8), 2079–2083 (1992)
10. Chen, K.: An integrated bit-serial 9-point median chip. In: Proceeding of the European Conference on Circuit Theory and Design, pp. 339–343 (1989)
11. Doggett, M., Meissner, M.: A memory addressing and access design for real time volume rendering. In: IEEE International Symposium on Circuits and Systems, ISCAS, vol. 4, pp. 344–347 (1999)

12. Dorati, A., Lamberti, C., Sarti, A., Baraldi, P., Pini, R.: Preprocessing for 3D echocardiography. Comput. Cardiol. 565–568 (1995)
13. Dupuy, D.E., Goldberg, S.N.: Image-guided radiofrequency tumor ablation: challenges and opportunities—part II. J. Vasc. Interv. Radiol. **12**(10), 1135–1148 (2001)
14. Fitch, J.P., Coyle, E.J., Gallagher, N.C.J.: Median filtering by threshold decomposition. IEEE Trans. Acoust. **32**(6), 1183–1188 (1984)
15. Gallegos-Funes, F.J., Ponomaryov, V.I.: Real-time image filtering scheme based on robust estimators in presence of impulsive noise. Real Time Imaging **10**(2), 69 (2004)
16. Gerig, G., Kubler, O., Kikinis, R., Jolesz, F.A.: Nonlinear anisotropic filtering of MRI data. IEEE Trans. Med. Imaging **11**(2), 221–232 (1992)
17. Gijbels, T., Six, P., Van Gool, L., et al.: A VLSI-architecture for parallel non-linear diffusion with applications in vision. In: Proc IEEE Workshop on VLSI Signal Processing, pp. 398–407 (1994)
18. Goldberg, N.S., Dupuy, D.E.: Image-guided radiofrequency tumor ablation: challenges and opportunities—part I. J. Vasc. Interv. Radiol. **12**(9), 1021–1032 (2001)
19. Haaga, J.R.: Interventional CT: 30 years' experience. Eur. Radiol. **15**, D116 (2005)
20. Hatirnaz, I., Gurkaynak, F.K., Leblebici, Y.: A compact modular architecture for the realization of high-speed binary sorting engines based on rank ordering. In: Proceedings of the IEEE International Symposium on Circuits and Systems, ISCAS, vol. 4, pp. 685–688 (2000)
21. Hawkes, D.J., McClelland, J., Chan, C., et al.: Tissue deformation and shape models in image-guided interventions: a discussion paper. Med. Image Anal. **9**(2), 163 (2005)
22. Hiasat, A.A., Al-Ibrahim, M.M., Gharaibeh, K.M.: Design and implementation of a new efficient median filtering algorithm. IEE Proc. Vis. Image Signal Process. **146**(5), 273–278 (1999)
23. Jiang, M., Crookes, D.: High-performance 3D median filter architecture for medical image despeckling. Electron. Lett. **42**(24), 1379 (2006)
24. Kar, B.K., Yusuf, K.M., Pradhan, D.K.: Bit-serial generalized median filters. In: Proceedings of the IEEE International Symposium on Circuits and Systems, ISCAS, vol. 3, pp. 85–88 (1994)
25. Karaman, M., Onural, L.: New radix-2-based algorithm for fast median filtering. Electron. Lett. **25**(11), 723–724 (1989)
26. Lee, C.L., Jen, C.-W.: Bit-sliced median filter design based on majority gate. IEE Proceedings, Part G: Circuits, Devices and Systems **139**(1), 63–71 (1992)
27. Lee, C.L., Jen, C.W.: A bit-level scalable median filter using simple majority circuit. In: Proceedings of IEEE International Symposium on VLSI Technology, Systems and Applications, 174–177 (1989)
28. Li, X., Chen, T.: Nonlinear diffusion with multiple edginess thresholds. Pattern Recognit. **27**(8), 1029–1037 (1994)
29. Oflazer, K.: Design and implementation of a single-chip 1D median filter. IEEE Trans. Acoust. **31**(5), 1164–1168 (1983)
30. Perona, P., Jitendra, M.: Scale-space and edge detection using anisotropic diffusion. IEEE Trans. Pattern Anal. Mach. Intell. **12**(7), 629–639 (1990)
31. Pfister, H.: Archtectures for real-time volume rendering. Future Gener. Comput. Syst. **15**(1), 1–9 (1999)
32. Pfister, H., Kaufman, A.: Cube-4-a scalable architecture for real-time volume rendering. In: Proceedings of the 1996 Symposium on Volume Visualization, pp. 47–54 (1996)
33. Roncella, R., Saletti, R., Terreni, P.: 70-MHz 2-um CMOS bit-level systolic array median filter. IEEE J. Solid State Circuits **28**(5), 530–536 (1993)
34. Rumpf, M., Strzodka, R.: Nonlinear diffusion in graphics hardware. In: Proceedings of EG/IEEE TCVG Symposium on Visualization, pp. 75–84 (2001)
35. Tabik, S., Garzon, E.M., Garcia, I., Fernandez, J.J.: Evaluation of parallel paradigms on anisotropic nonlinear diffusion. Eur. Par. Parallel Process. **4128**, 1159 (2006)
36. Viola, I., Kanitsar, A., Groller, M.E.: Hardware-based nonlinear filtering and segmentation using high-level shading languages. IEEE Vis. 309 (2003)
37. Whitaker, R.T., Pizer, S.M.: A multi-scale approach to nonuniform diffusion. CVGIP Image Underst. **57**(1), 99–110 (1993)
38. Wiehler, K., Heers, J., Schnorr, C., Stiehl, H.S., Grigat, R.-R.: A one-dimensional analog VLSI implementation for nonlinear real-time signal preprocessing. Real Time Imaging **7**(1), 127–142 (2001)

Author Biographies

Omkar Dandekar Omkar Dandekar received the B.E. degree in Biomedical Engineering from the University of Mumbai, India in 2000, and the M.S. degree in Electrical Engineering from the Ohio State University, Columbus, in 2004. He is currently a doctoral candidate in the department of Electrical and Computer Engineering at the University of Maryland, College Park. He has worked as a graduate research assistant at the Cleveland Clinic Foundation. His primary interests include medical imaging, digital VLSI design, and hardware acceleration of image processing algorithms. Currently, his research work is focused on real-time 3D imaging and advanced image processing and analysis for image-guided interventions.

Carlos Castro-Pareja Carlos R. Castro-Pareja received the B.Sc. degree in electrical engineering from the Pontificia Universidad Católica del Perú, Lima, Peru, in 1999, and the M.Sc. and Ph.D. degrees in Electrical Engineering from the Ohio State University, Columbus, in 2001 and 2004. He is currently employed at Intel Corp. His interests include development and hardware acceleration of image processing algorithms.

Raj Shekhar Raj Shekhar, Ph.D., is an Assistant Professor of Diagnostic Radiology, Bioengineering, and Electrical and Computer Engineering at the University of Maryland, Baltimore and College Park. He previously served as a Staff Scientist at the Cleveland Clinic and as a Senior Engineer at Picker International (now Philips Medical Systems). Dr. Shekhar received his doctorate in Biomedical Engineering from the Ohio State University in 1997. Dr. Shekhar's research interests are medical image processing, real-time computing, 3D ultrasound, and image-guided interventions. Dr. Shekhar has authored over 50 scientific papers, including over 20 peer-reviewed articles. He also holds three US patents.

FPGA-Accelerated Deformable Image Registration for Improved Target-Delineation During CT-Guided Interventions

Omkar Dandekar, *Member, IEEE*, and Raj Shekhar, *Member, IEEE*

Abstract—**Minimally invasive image-guided interventions (IGIs) are time and cost efficient, minimize unintended damage to healthy tissue, and lead to faster patient recovery. With the advent of multislice computed tomography (CT), many IGIs are now being performed under volumetric CT guidance. Registering pre- and intraprocedural images for improved intraprocedural target delineation is a fundamental need in the IGI workflow. Earlier approaches to meet this need primarily employed rigid body approximation, which may not be valid because of nonrigid tissue misalignment between these images. Intensity-based automatic deformable registration is a promising option to correct for this misalignment; however, the long execution times of these algorithms have prevented their use in clinical workflow. This article presents a field-programmable gate array-based architecture for accelerated implementation of mutual information (MI)-based deformable registration. The reported implementation reduces the execution time of MI-based deformable registration from hours to a few minutes. This work also demonstrates successful registration of abdominal intraprocedural noncontrast CT (iCT) images with preprocedural contrast-enhanced CT (preCT) and positron emission tomography (PET) images using the reported solution. The registration accuracy for this application was evaluated using 5 iCT-preCT and 5 iCT-PET image pairs. The registration accuracy of the hardware implementation is comparable with that achieved using a software implementation and is on the order of a few millimeters. This registration accuracy, coupled with the execution speed and compact implementation of the reported solution, makes it suitable for integration in the IGI-workflow.**

***Index Terms*—Computer tomography (CT)-guided interventions, field-programmable gate arrays (FPGA), image registration, mutual information.**

I. Introduction

IMAGE-GUIDED interventions (IGIs), including surgeries, biopsies, and therapies, have the potential to improve patient care compared with conventional open and invasive procedures by enabling new and faster procedures, minimizing damage to healthy tissues, improving the effectiveness of procedures, producing fewer complications, and allowing for clinical intervention at a distance. Continuous 3-D imaging and visualization for intraprocedural navigation, critically important to the success of IGIs, has been technologically difficult until recently. However, the emergence of multislice computed tomography (CT) technology now provides the opportunity to achieve necessary imaging speed, coverage of the operative field (about 4–8 cm), and high-resolution (up to 0.625 mm) intraprocedural imaging. As a result, many IGIs are now being routinely carried out under volumetric CT guidance [1]–[6].

The efficiency and efficacy of IGIs is critically dependent on accurate and precise target identification and localization. Lack of clear target delineation could lead to lengthy procedures, larger than necessary safety margins, and unintended damage to healthy tissue—factors that undermine the very motivation behind IGIs. Intraprocedural imaging techniques provide a rich source of accurate spatial information that is crucial for navigation but often suffer from poor target definition from background healthy and/or benign tissue. As in most clinical protocols, IGIs are preceded by one or more preprocedural images, containing additional information, such as contrast-enhanced structures or functional details such as metabolic activity (Fig. 1), which are used for diagnosis, treatment/navigation planning, etc. Combining this functional and/or contrast information with intraprocedural morphological and spatial information, through image registration between pre- and intraprocedural images, has been shown to improve the intraprocedural target delineation [7]–[13].

Deformable image registration techniques can compensate for both local deformation and large-scale tissue motion and are the ideal solution for achieving the aforementioned image registration. Some studies, in particular, have independently underlined the importance of deformable registration and/or soft tissue modeling for IGIs [14], [15]. However, despite their advantages, deformable registration algorithms are seldom used in current clinical practice. The large number of degrees of freedom that these algorithms employ makes them extremely computationally intensive. On a modern workstation most deformable registration algorithms can take several hours, which is clearly unacceptable for IGIs requiring *on-demand* performance. As a result, most earlier reported techniques for aligning preprocedural and intraprocedural images employ rigid body approximation, which is often not valid because of underlying nonrigid tissue deformation. In addition, some of these techniques are not retrospective (i.e., they require

Manuscript received June 12, 2007; revised September 4, 2007. This work was supported in part by the U.S. Department of Defense (TATRC) under Grant DAMD17-03-2-0001. This paper was recommended by Associate Editor R. Butera.

O. Dandekar is with the Department of Electrical and Computer Engineering, University of Maryland, College Park, MD 20742 USA and also with the Department of Diagnostic Radiology and Nuclear Medicine, University of Maryland, Baltimore, MD 21201 USA (e-mail: dandekar@ieee.org).

R. Shekhar is with the Department of Diagnostic Radiology and Nuclear Medicine, University of Maryland, Baltimore, MD 21201 USA (e-mail: rshekhar@umm.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TBCAS.2007.909023

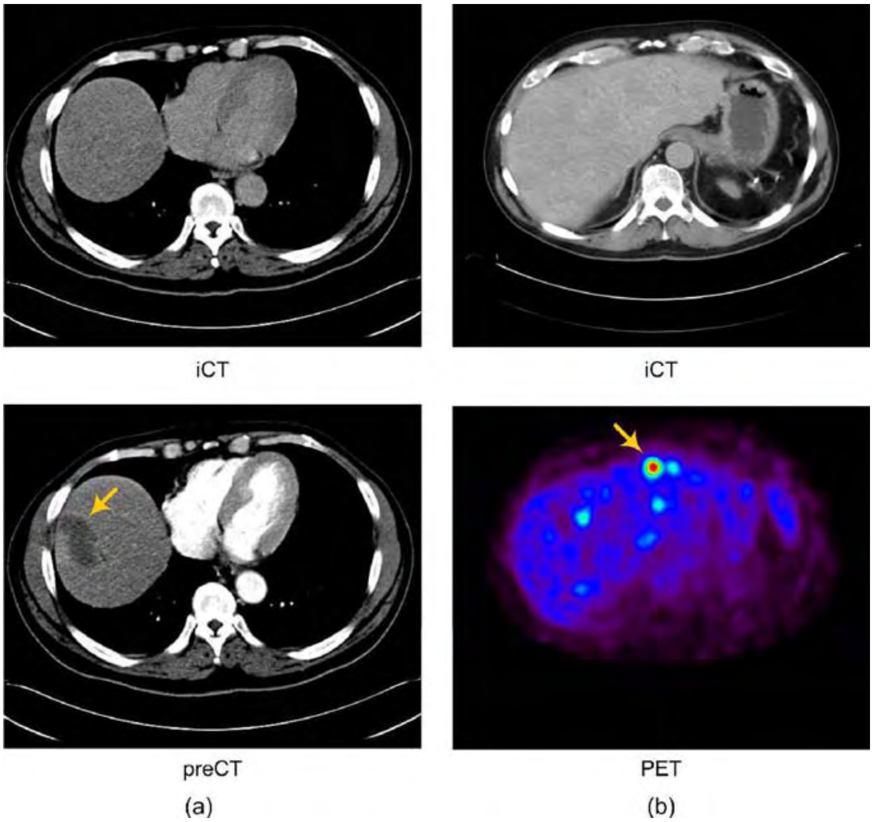


Fig. 1. Comparison of intraprocedural and preprocedural images for the same subject. Image (a) shows an example of intraprocedural noncontrast CT (iCT) and preprocedural contrast-enhanced CT (preCT) image pair; and (b) shows an example of an iCT and positron emission tomography (PET) image pair. Preprocedural images typically offer better target delineation, as indicated by the overlaid arrows.

some advanced planning at the time of preprocedural imaging), which further limits their applicability.

Mutual information (MI)-based deformable registration has been shown to be effective in multimodality image registration because of the robustness of the similarity measure [16]. Moreover, MI-based image registration is automatic and completely retrospective because it uses image intensities to achieve the alignment. Hierarchical volume subdivision–based image registration techniques are inherently faster than most other deformable registration techniques and are more amenable to hardware acceleration. Walimbe and Shekhar [17], [18] have earlier reported an MI-based deformable registration algorithm that utilizes volume subdivision. This algorithm has been used and rigorously validated in the context of positron emission topography (PET)-CT registration [19]. This clinical validation has demonstrated the registration accuracy of the aforementioned algorithm to be comparable to a group of clinical experts and the mean registration accuracy for the abdominal region to be superior to an earlier reported free-form deformation (FFD)-based technique [20]. This algorithm is theoretically general and has been shown to be effective for various applications employing multimodal deformable registration [21]–[26]. Although computationally efficient, software implementation of this algorithm can take several hours, which is still slow for direct integration into the IGI workflow. It is, therefore, necessary to accelerate this algorithm and reduce the processing time to the order of minutes and ultimately to seconds for its assimilation into clinical workflow. Although, accelerated implementations of MI-based deformable registration algorithms using very large multiprocessor clusters have been proposed earlier [27]–[30], these solutions may not be cost effective and are unlikely to be suitable for clinical deployment.

The current work presents a novel field-programmable gate array (FPGA)-based accelerated implementation of the aforementioned deformable registration algorithm, specially geared toward improving target delineation during CT-guided interventions. The reported solution provides a speedup of about 40 for MI calculation, thus reducing the deformable registration time from hours to minutes. We demonstrate fast and successful registration of abdominal intraprocedural noncontrast CT (iCT) with preprocedural contrast-enhanced CT (preCT) and PET images using the reported implementation. The registration accuracy for this application was evaluated based on validation using expert-identified anatomical landmarks in 5 iCT-preCT and 5 iCT-PET image pairs. The registration accuracy of the hardware implementation is comparable with that using a software implementation and is on the order of a few millimeters. This registration accuracy coupled with the execution speed and compact implementation of the reported solution makes it suitable for integration in the IGI workflow.

II. Registration Algorithm

Hierarchical volume subdivision-based deformable image registration techniques are inherently faster than most de-

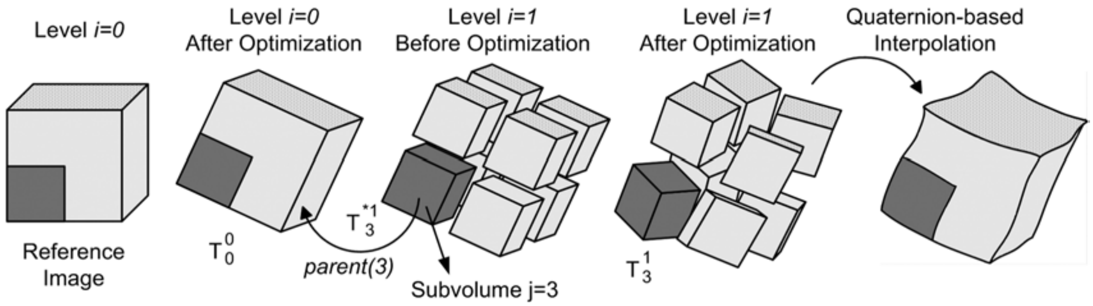

Level i=0
Level i=0 After Optimization
Level i=1 Before Optimization
Level i=1 After Optimization
Quaternion-based Interpolation
Reference Image
T_0^0
T_3^{*1}
parent(3)
Subvolume j=3
T_3^1

Fig. 2. Pictorial representation of hierarchical volume subdivision-based deformable image registration and associated notation.

formable registration techniques (e.g., FFD-based techniques) and are more amenable to acceleration through hardware implementation. 3-D image registration using volume subdivision has been proposed earlier, but the earlier implementations were limited to a local translation-based model. Walimbe and Shekhar [17], [18] enhanced this model by incorporating local rotations and reported a quaternion-based scheme for interpolating multiple 3-D rigid-body transformations for deformable registration using the volume subdivision approach. For a pair of images, one treated as reference and the other as floating, this algorithm performs deformable registration using a series of hierarchical, locally rigid-body registrations. The six-parameter rigid registration at each level is optimized by maximizing the MI between the reference and floating images (RI and FI, respectively). This hierarchical registration scheme is shown in Fig. 2.

The initial optimal rigid alignment (at the root level) between RI and FI can be represented using a transformation matrix T_0^0 (where T_j^i represents the cumulative optimal transformation at level i for subvolume j). Next, the algorithm uses a hierarchical octree-based subdivision scheme. At each subdivision level i, the RI is divided into 8^i subvolumes, numbered from 0 to $8^i - 1$. Each of these subvolumes is then individually registered with the FI, under transformation range constraints derived from the transformation of its parent subvolume at the earlier level $T^{i-1}_{\mathrm{parent}(j)}$. The notation $\mathrm{parent}(j)$ refers to the subvolume at the previous subdivision level $i-1$, which contains the current subvolume j. For example, at the root level $(i = 0)$, there is a single subvolume (entire image) numbered $j = 0$. After one level of subdivision $(i = 1)$, there will be eight subvolumes numbered from $j = 0$ to $j = 7$. At level $i = 1$, $\mathrm{parent}(3)$ refers to subvolume numbered 0 at level $i = 0$, because it contains subvolume $j = 3$ at the current level $(i = 1)$ of subdivision (Fig. 2). The optimal alignment of the subvolume j within the FI is also determined by maximizing MI under a six-parameter rigid-body transformation model.

Volume subdivision and subvolume registration continue until the voxel count for an individual subvolume remains above a predefined limit (usually 16^3) to yield a statistically significant similarity measure. Thus, this algorithm achieves hierarchical refinement of the localized matching between RI and FI. The final cumulative nonrigid alignment between the image pairs is computed by quaternion-based direct interpolation of the individual subvolume transformations at the final subdivision level.

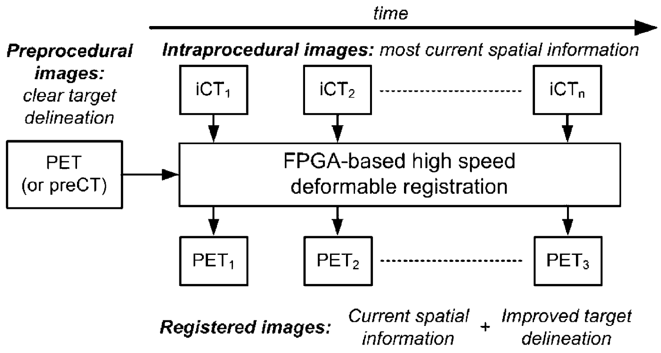

time
Preprocedural images: clear target delineation
Intraprocedural images: most current spatial information
iCT1
iCT2
iCTn
PET (or preCT)
FPGA-based high speed deformable registration
PET1
PET2
PET3
Registered images:
Current spatial information
+
Improved target delineation

Fig. 3. Integration of FPGA-based high-speed deformable registration in the IGI workflow to provide improved intraprocedural target delineation.

A. Calculating MI for a Subvolume

Registration of a subvolume during the hierarchical refinement process is based on maximization of the MI, which is a statistical measure. With progressive subdivision, subvolumes at every level become increasingly smaller. The mutual histogram (MH) corresponding to an individual subvolume becomes sparse, thus rendering MI unreliable. The aforementioned algorithm addresses this issue by using the MH of the entire image (all the subvolumes) to calculate MI during the registration of a subvolume. The contribution of the current subvolume k at level i to the MH is computed under the current candidate transformation T_k^{*i} (T^* denotes a candidate transformation during the optimization process). The contribution to the MH from the rest of the subvolumes remains constant during this registration process and is derived from their parent subvolumes. Thus, MI is computed over the entire image with local variations corresponding to the subvolume under optimization. Equations (1)–(3) summarize this process. The function $\underset{j=k}{\mathrm{Accumulate}}(T)$, contributes to the MH using the voxels in a given subvolume k, using the mapping provided by the given transformation T. The detailed description of this deformable registration algorithm can be found in [17]

$$\mathrm{MH}^i_{\mathrm{Total}_k} = \mathrm{MH}^i_{\mathrm{Subvolume}_k} + \mathrm{MH}^i_{\mathrm{Rest}_k} \tag{1}$$

$$\mathrm{MH}^i_{\mathrm{Subvolume}_k} = \underset{j=k}{\mathrm{Accumulate}}\left(T_j^{*i}\right) \tag{2}$$

$$\mathrm{MH}^i_{\mathrm{Rest}_k} = \underset{\forall\, j, j\neq k}{\mathrm{Accumulate}}\left(T^{i-1}_{\mathrm{parent}(j)}\right). \tag{3}$$

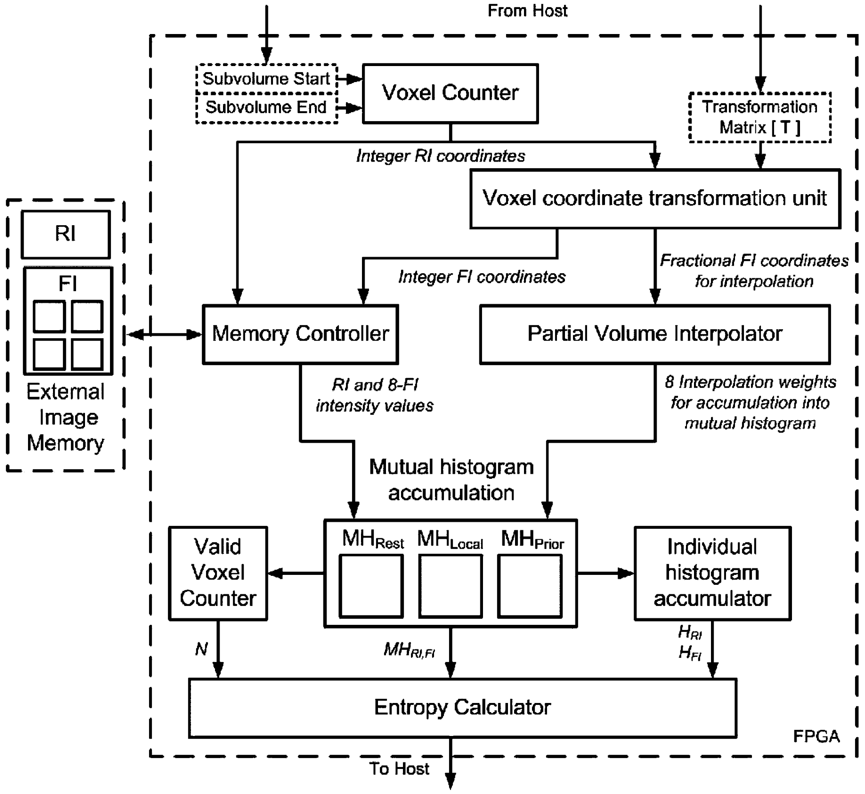

From Host
Subvolume Start
Subvolume End
Voxel Counter
Transformation Matrix [T]
Integer RI coordinates
Voxel coordinate transformation unit
RI
FI
External Image Memory
Integer FI coordinates
Fractional FI coordinates for interpolation
Memory Controller
Partial Volume Interpolator
RI and 8-FI intensity values
8 Interpolation weights for accumulation into mutual histogram
Mutual histogram accumulation
MH_{Rest} MH_{Local} MH_{Prior}
Valid Voxel Counter
Individual histogram accumulator
N
$MH_{RI,FI}$
H_{RI}
H_{FI}
Entropy Calculator
FPGA
To Host

Fig. 4. Block diagram of the reported architecture.

III. Methods

Accurate and high-speed implementation of multimodality deformable image registration can enable its integration in IGI workflow as shown in Fig. 3. One of the benefits of this integration is better intraprocedural target delineation, which may lead to improved procedure outcomes. An important step toward meeting this goal is acceleration of deformable image registration. The aforementioned algorithm uses MI as a measure of image similarity. Registration through maximization of MI attempts to find the transformation that best aligns an FI with an RI. MI-based registration typically requires thousands of iterations (MI evaluations), depending on the image complexity and the degree of initial misalignment between the images. Repeated MI computation, which requires accessing both the images (RI and FI), is memory access intensive, and the memory access in the FI is completely governed by the transformation applied. This operation, therefore, does not benefit from the cache-based memory architectures present in most modern PCs (the caches are too small to fit 3-D images). Because memory speed has not evolved at the same rate as microprocessor speed, introduction of faster microprocessors is not expected to significantly speed up image registration. Thus, a factor limiting the performance of software implementations is calculating MI for different candidate transformations. Constructing the MH for a given transformation is a critical step in all MI-based image registration algorithms. Castro-Pareja *et al.* [31] have shown that, for large images, accumulating the MH can take up to 99.9% of the total MI calculation time in software. Our efforts for acceleration of this algorithm consequently are targeted toward optimized and pipelined implementation of MH accumulation and MI calculation.

Section III-A describes the architecture for high-speed implementation of the volume subdivision-based deformable registration algorithm in detail. Section III-B describes our strategy to evaluate the accuracy of deformable registration performed using this high-speed implementation.

A. Architecture

MI-based image registration can be thought of as an optimization problem of finding the best alignment between two images. During the execution of the algorithm, the optimization process is executed from the host workstation. This host provides a candidate transformation, while the reported FPGA-based solution applies it to the images and performs corresponding MI computation. The computed MI value is then further used by the host to update the candidate transformation and eventually find the optimal deformable alignment between the RI and the FI. Fig. 4 shows the top-level block diagram of the reported architecture customized for accelerated implementation of volume subdivision-based image registration. The important modules in this design are described below.

1) Voxel Counter: Calculation of MI requires processing each voxel in the RI. In addition, because the implemented algorithm is based on volume subdivision, RI voxels within a 3-D neighborhood corresponding to an individual subvolume must be processed sequentially. The host programs the FPGA-based MI calculator with subvolume start and end addresses and the voxel counter computes the address corresponding to each voxel within that subvolume in z-y-x order.

2) Coordinate Transformation: The initial step in MI calculation involves applying a candidate transformation $\left(T_j^{*i}\right)$, to each voxel coordinate $(\vec{v}_r)$ in a subvolume j of the RI to find the corresponding voxel coordinates in the FI $(\vec{v}_f)$. This is mathematically represented as shown in (4). Because the algorithm is linear at every subvolume, this is implemented using the six-parameter rigid transformation model

$$\vec{v}_f = T_j^{*i} \cdot \vec{v}_r. \tag{4}$$

This transformation model is represented using a 4 × 4 matrix. The host calculates this matrix based on the current candidate transformation provided by the optimization routine and sends it to the MI calculator. Fixed-point representation is used to store the individual elements of this matrix. The coordinate transformation is accomplished by a simple matrix multiplication.

3) Partial Volume Interpolation: The coordinates mapped in the FI space $(\vec{v}_f)$ do not normally coincide with a grid point (integer location), thus requiring interpolation. Nearest neighbor and trilinear interpolation schemes have been traditionally used for this purpose; however, partial volume (PV) interpolation, introduced by Maes *et al.* [16] has been shown to provide smooth changes in the histogram values with small changes in transformation. The reported architecture consequently implements PV interpolation as the choice of interpolation scheme. $\vec{v}_f$, in general, will have both fractional and integer components and will land within an FI neighborhood of size 2 × 2 × 2. The interpolation weights required for the PV interpolation are calculated using the fractional components of $\vec{v}_f$. Fixed-point arithmetic is used to compute these interpolation weights. The corresponding floating voxel intensities are fetched by the image controller in parallel using the integer component of $\vec{v}_f$. The image controller also fetches the voxel intensity corresponding to $\vec{v}_r$. The MH then must be updated for each pair of reference and floating voxel intensities (8 in all), using the corresponding weights computed by the PV interpolator.

4) Image Memory Access: The size of most 3-D images constrains the use of high-speed memory internal to the FPGA for their storage. Between the two images, the RI has more relaxed access requirements, because it is accessed in a sequential manner (in z-y-x order). This kind of access benefits from burst accesses and memory caching techniques, allowing the use of modern dynamic random access memories (DRAM) for image storage. For the architecture presented, both the RI and FI are stored in separate logical partitions of the same DRAM module. Because the access to the RI is sequential and predictable, the architecture uses internal memory to cache a block of RI voxels. Thus, during the processing of that block of RI voxels, the image controller has parallel access to both RI and FI voxels. The RI voxels are fetched from the internal FPGA memory, whereas the FI voxels are fetched directly from the external memory.

The FI, however, must be accessed randomly (depending on the current transformation T_j^{*i}), and 8 FI voxels (2 × 2 × 2 neighborhood) must be fetched for every RI image voxel that is being processed. To meet this memory access requirement, the reported architecture employs a memory addressing scheme similar to the cubic addressing technique reported in the context of volume rendering [32]. A salient feature of this technique is that it allows simultaneous access to the entire 2 × 2 × 2 voxel neighborhood. The reported architecture implements this technique by storing four copies of the FI and taking advantage of the burst mode accesses native to modern DRAMs. The image voxels are arranged sequentially such that, performing a size 2 burst fetches two adjacent 2 × 2 neighborhood planes, thus making the entire neighborhood available simultaneously. The image intensities of this neighborhood are then further used for updating the MH.

5) Updating the Mutual Histogram: For a given RI voxel *RV*, there are eight intensity pairs $(RV, FV_0 : FV_7)$ and corresponding interpolation weights. Because the MH must be updated (read-modify-write) at these eight locations, this amounts to 16 accesses to MH memory for each RI voxel. This high memory access requirement is addressed by using the high-speed, dual-ported memories internal to the FPGA to store the MH. The operation of updating the MH is pipelined and, hence, read-after-write (RAW) hazards could arise if consecutive transactions attempt to update identical locations within the MH. The reported design addresses this issue by introducing *preaccumulate buffers*, which aggregate the weights from all the conflicting transactions. Thus, all the transactions leading to a RAW hazard are converted into a single update to the MH thereby eliminating any RAW hazards.

While the MH is being computed, the individual histogram accumulator unit computes the histograms for the RI and the FI. These individual histograms are also stored using internal, dual-ported memories. The valid voxel counter module keeps track of the number of valid voxels accumulated in the MH, and calculates its reciprocal value. The resulting value is then used by the entropy calculation unit for calculating the individual and joint probabilities.

Calculating $\mathrm{MH}_{\mathrm{Rest}}$*:* During the optimization process of finding the best alignment for a given subvolume k at level i, $\mathrm{MH}_{\mathrm{Total}_k}$ is computed as shown in (1). For this calculation, it is necessary to compute the contribution of the remaining subvolumes to $\mathrm{MH}_{\mathrm{Total}_k}$ using the registration information at the previous level of subdivision $(T_{\mathrm{parent}(j)}^{i-1}, \ \forall\ j \neq k)$. This process must be repeated for every subvolume at the current level i, and the contents of $\mathrm{MH}_{\mathrm{Rest}_k}$ will be different every time depending on the subvolume under consideration. Computing $\mathrm{MH}_{\mathrm{Rest}_k}$ from scratch for every iteration of each subvolume will not be efficient. To avoid this repeated calculation, we introduce MH buffers ($\mathrm{MH}_{\mathrm{Prior}}$, $\mathrm{MH}_{\mathrm{Rest}}$, and $\mathrm{MH}_{\mathrm{Local}}$), which store the previous-level MH and partial MH during various stages of the algorithm. A flow diagram depicting the interplay between these MH buffers during various steps of calculating $\mathrm{MH}_{\mathrm{Rest}}$ is shown in Fig. 5, and the detailed description is provided here.

At a given level i, $\mathrm{MH}_{\mathrm{Prior}}$ contains the MH for the entire image, which is computed using all the subvolumes at the earlier level $i-1$ and corresponding transformations T_j^{i-1}. At the beginning of the registration of each subvolume k at the current level i, $\mathrm{MH}_{\mathrm{Local}}$ is cleared (thus, all its entries are set to 0). Next, the transformation of its parent from the previous level, $T_{\mathrm{parent}(k)}^{i-1}$, is applied to the current subvolume k and the resultant MH is accumulated in $\mathrm{MH}_{\mathrm{Local}}$. $\mathrm{MH}_{\mathrm{Local}}$ now con-

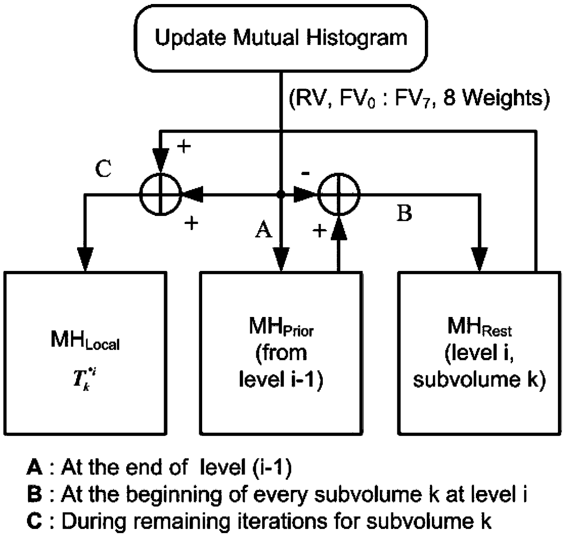

Update Mutual Histogram
(RV, FV0 : FV7, 8 Weights)
C
+
+
-
A
+
B
MHLocal
T_k^{*i}
MHPrior
(from
level i-1)
MHRest
(level i,
subvolume k)
A : At the end of level (i-1)
B : At the beginning of every subvolume k at level i
C : During remaining iterations for subvolume k

Fig. 5. A flow diagram of steps involved in calculating $\mathrm{MH}_{\mathrm{Rest}}$.

tains the contribution of the subvolume k to $\mathrm{MH}_{\mathrm{Prior}}$. $\mathrm{MH}_{\mathrm{Local}}$ is then subtracted from $\mathrm{MH}_{\mathrm{Prior}}$, and the resultant MH is stored in the buffer $\mathrm{MH}_{\mathrm{Rest}}$. This step is mathematically equivalent to computing $\mathrm{MH}_{\mathrm{Rest}_k}$ [as in (3)] for the subvolume k. For every subsequent optimization iteration involving this subvolume, $\mathrm{MH}_{\mathrm{Local}}$ is initially cleared. The candidate transformation T_k^{*i} is then applied to the current subvolume (k) only, and the resultant histogram (contribution of this subvolume) is accumulated in $\mathrm{MH}_{\mathrm{Local}}$. It is then added to $\mathrm{MH}_{\mathrm{Rest}_k}$, to form the MH for the entire image [$\mathrm{MH}_{\mathrm{Total}_k}$ in (1)] for the current optimization step. This final histogram is then further used for computing the image similarity measure (MI) corresponding to the current transformation T_k^{*i}. This process is repeated for all the subvolumes at the current level. At the end of each level (after optimizing all the subvolumes at that level), the MH for the entire image is computed using the updated transformations $T_k^i, \forall\, k$ and is stored in $\mathrm{MH}_{\mathrm{Prior}}$, which will then subsequently be used at the next level of subdivision, $i+1$.

6) Entropy Calculation: The final step in MI calculation is to compute joint and individual entropies using the joint and individual probabilities, respectively. To calculate entropy, it is necessary to evaluate the function $f(p) = p \cdot \ln(p)$ for all the probabilities. As probability p takes values between $[0, 1]$, the corresponding range for the function $f(p)$ is $[-e^{-1}, 0]$. Thus, $f(p)$ has a finite dynamic range and is defined for all values of p. Several methods for calculating logarithmic functions in hardware have been reported earlier [33]–[35], but of particular interest is the multiple lookup table (LUT)-based approach introduced by Castro-Pareja *et al.* [36]. This approach minimizes the error in representing $f(p)$ for a given number and size of LUTs and, hence, is accurate and efficient. Moreover, this approach preserves the shape of the MI curve and the location of the extrema and thus does not significantly affect the outcome of the optimization process. Following this approach, the reported design implements $f(p)$ using multiple LUT-based piecewise polynomial approximation.

B. Validation of Deformable Image Registration

The reported architecture is designed to accelerate the volume subdivision-based deformable registration algorithm and reduce the registration time to the order of minutes. This architecture is thus capable of offering performance that is superior in terms of execution speed (Table II) to a software implementation. For medical applications such as the one reported here, however, the ability of a proposed solution to achieve the desired level of accuracy is of paramount importance. It is, therefore, necessary to evaluate the registration accuracy achieved by this accelerated implementation and compare it with that obtained by a reference software implementation. Because the reported design is targeted toward CT-guided interventions, we focus on image registration of iCT with preCT and PET, which are the two most common preprocedural imaging modalities in the context of CT-guided interventions.

We evaluate the accuracy of iCT-preCT and iCT-PET registration by comparing the alignment of several anatomic landmarks as predicted by the algorithm (both software and hardware implementations) against a reference. Because of the lack of a gold standard for most applications involving deformable registration, we assume the ability of clinical experts to identify landmarks in iCT, preCT, and PET images as a suitable reference. To account for slight observer variability, we compute the centroid of the locations identified by multiple experts for a given landmark and use that as a reference. We then compare the landmarks predicted by the software and hardware implementations of the automatic deformable registration algorithm against this reference. The target registration error (TRE) at these landmarks is then used as a metric to compare the registration accuracy of the high-speed FPGA-based implementation with the software implementation.

This validation was performed using five abdominal iCT-preCT and five abdominal iCT-PET image pairs. The typical image size for iCT and preCT was $256 \times 256 \times 200-350$ with a voxel size of 1.4–1.7 mm × 1.4–1.7 mm × 1.5 mm, whereas the typical image size for PET was $128 \times 128 \times 154 - 202$ voxels with a voxel size of 5.15 mm × 5.15 mm × 5.15 mm. Three clinical experts, experienced in interpreting CT and PET images, were involved in the validation procedure. Each expert was asked to identify and mark homologous anatomic landmarks in all the images from a list of 20 well-described landmarks. Examples of anatomic landmarks included dome of the liver, upper and lower tips of the kidneys, top of the spleen, and so forth. Comparable TRE at these landmarks for both software and hardware implementations would indicate no significant difference between registration accuracy of software and the reported FPGA-based high-speed implementations of the deformable registration algorithm.

IV. Implementation and Results

The reported architecture was implemented using an Altera Stratix II EP2S180F1508C4 FPGA (Altera Corporation, San Jose, CA) in a PCI prototyping board (DN7000K10PCI) manufactured by the Dini Group (La Jolla, CA). The board-featured 1-GB double-data-rate (DDR2) DRAM in a small-outline dual-inline memory module (SoDIMM) external to the FPGA.

TABLE I
COMPARISON OF MUTUAL INFORMATION CALCULATION TIME FOR SUBVOLUMES AT VARIOUS LEVELS IN VOLUME SUBDIVISION-BASED DEFORMABLE REGISTRATION ALGORITHM

Subdivision level	Subvolume size	Mutual information calculation time (ms)		Speedup
		Software implementation	FPGA-based implementation	
0[a]	256 × 256 × 256	9410	225.42	41.74
1	128 × 128 × 128	1209	30.19	40.05
2	64 × 64 × 64	166	4.16	39.90
3	32 × 32 × 32	18	0.78	23.08
4	16 × 16 × 16	10	0.46	21.74

[a]This corresponds to rigid registration between the reference and floating images.

This memory was used to store the RI and the FI. The board provided a 64-bit bus interface between the memory and the FPGA running at 200 MHz clock speed. The architecture was designed using VHSIC hardware description language (VHDL) and synthesized using Altera Quartus II 6.1. The memory controller was also implemented using VHDL and was built around the DDR2 DRAM controller megacore supplied with Altera Quartus II. All the modules in the architecture were custom designed using VHDL, per the design description in Section III-A. Functional verification and postsynthesis timing simulation for the entire system were performed using Modelsim SE 6.2 (Mentor Graphics, San Jose, CA). For this purpose, DDR2 DRAM simulation models provided by Micron (www.micron.com) were used. The resource availability of the target FPGA, primarily internal memory, which is used for MH accumulation, limited the synthesis of the reported design to support 7-bit images and, consequently, all results in this section are presented for 7-bit images. In comparison, the software implementation uses 8-bit images. In spite of this difference, the registration accuracy for both these implementations is similar, which further confirms the findings of Studholme *et al.* [37]. The iCT, preCT, and PET images were converted to 8 and 7 bits, respectively, for software and FPGA-based implementations. This conversion was performed using adaptive reduction in intensity levels described in [38]. The converted iCT and preCT images were then preprocessed using 3-D anisotropic diffusion filtering. No preprocessing steps were used for the PET images. Previously reported real-time implementation of anisotropic diffusion filtering can facilitate this preprocessing without adding any significant latency to IGI workflow [39]. For the hardware implementation, the RI was stored in its native sequential format while four interleaving copies of the FI were arranged in memory to facilitate cubic addressing. This arrangement allows simultaneous access to an entire 3-D neighborhood within the FI, as described earlier. The execution speed of the reported architecture was obtained from postsynthesis timing measurements using the entire system.

The design achieved a maximum internal frequency of 200 MHz, with a 100 MHz RI processing rate. The coordinate transformation, PV interpolation, and MH accumulation operations were implemented using fixed-point representation. Entropy calculation was implemented using the four-LUT, first-order polynomial configuration and also employed 32-bit fixed-point representation. As shown previously [36], the maximum error in the entropy calculation using this configuration was on the order of 10^{-8}.

A. Execution Speed

The reported architecture is targeted toward accelerating the calculation of MI for a hierarchical volume subdivision-based deformable registration algorithm. During the execution of this algorithm, MI must be repeatedly calculated under a candidate transformation for every subvolume at every level of subdivision. Moreover, as described earlier, $\mathrm{MH}_{\mathrm{Rest}}$ must be calculated once for every subvolume. To analyze the speedup offered by the reported FPGA-based solution for calculation of MI, we compare its calculation time with that of a software (C++) implementation running on an Intel Xeon 3.6 GHz workstation with 2 GB of RAM. Table I details this performance comparison for a iCT-preCT image pair with dimensions of 256 × 256 × 256. The last column of the table shows the speedup offered by the reported solution over the software implementation for a given level of subdivision. The time to calculate MI primarily depends on the size of the subvolume and is independent of the imaging modality and voxel size. This calculation time, however, may vary slightly based on the actual value of the transformation used to calculate MI. These variations are caused by differences in access patterns to the FI memory under different transformation values. To compensate for this effect and report average MI calculation time at a given level of subdivision, we measured the MI calculation times for a subvolume using 100 randomly generated transformations (within the range of ±30 voxels for translations and ±20° for rotations). The same set of transformations was used for both software and hardware implementations. The MI calculation time reported in Table I is averaged over all the transformations. Hardware timings reported in Table I also include the communication time required for writing the transformation matrix and reading back the calculated entropy values between the host and the MI calculator. Furthermore, consistent with the scenario during the execution of the registration algorithm, the time to compute $\mathrm{MH}_{\mathrm{Rest}}$ (once per subvolume) is also included in the hardware and software MI calculation time.

Table II compares the total execution time for deformable registration using a software (C++) implementation running on an Intel Xeon 3.6 GHz workstation with 2 GB of RAM against the reported FPGA-based design. This execution time was measured for deformable registration between five iCT-preCT and five iCT-PET image pairs, described earlier. The maximum number of global and local iterations was set to 200 and 100, respectively. The same optimization algorithm was used for both the software and hardware implementations, and volume subdivision continued until the subvolume size was larger

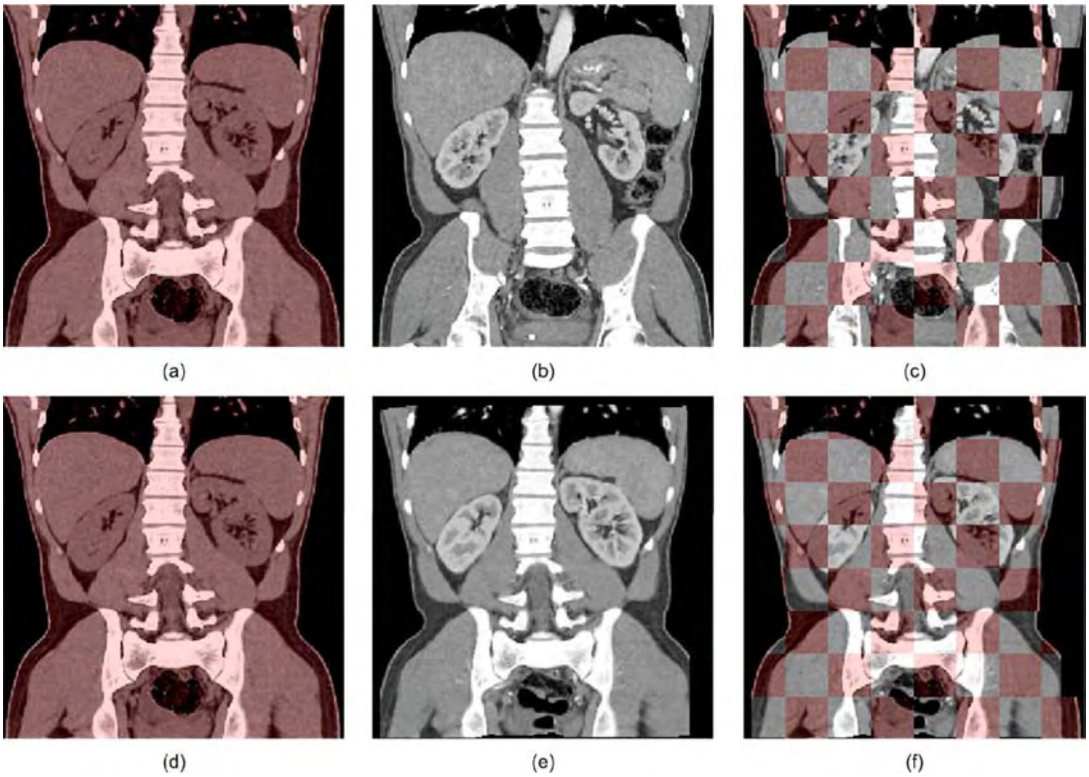


Fig. 6. Qualitative validation of deformable registration between iCT and preCT images performed using the reported solution: Images (a) and (d) are identical and show a coronal slice from an iCT image; (b) shows a corresponding coronal slice from a preCT image; (c) shows fusion of (a) and (b) using a checkerboard pattern, indicating evident structural misalignment; (e) shows a corresponding coronal slice from the preCT image registered to the iCT image; and (f) shows fusion of (d) and (e) using a checkerboard pattern, indicating improved alignment of structures after deformable registration.

TABLE II
EXECUTION TIME OF DEFORMABLE REGISTRATION

Image pair used for registration	Execution time (s)		Speedup
	Software implementation	FPGA-based implementation	
iCT-preCT	11520	371	31.05
iCT-PET	11146	339	32.88

than 16^3. For each image modality pair, the execution time reported is the average of execution times of the five cases. The execution time of intensity-based image registration is directly proportional to the size of the RI. iCT image was used as the RI for both the image modality pairs and, hence, the execution time for deformable registration is similar for these two image modalities, despite the fact that PET images are smaller than preCT images ($128 \times 128 \times 154 - 202$ and $256 \times 256 \times 200 - 350$, respectively). For both image modality pairs, the reported solution provided a speedup of about 30 over an equivalent software implementation and reduced the execution time from hours to a few minutes.

B. Registration Accuracy

Acceleration of deformable registration as offered by the reported design is critical for the time-sensitive nature of IGIs; however, the accuracy of the registration process is of equal importance. Because the reported design is targeted toward CT-guided interventions, we focus on evaluating the accuracy of registration of iCT with preCT and PET. As described, the validation of deformable registration was performed using five iCT-preCT and five iCT-PET image pairs.

Fig. 6 shows an example of deformable registration between a pair of iCT and preCT images. This registration was performed using the reported FPGA-based solution. The initial seed alignment between these images was obtained using a single-click operation to compensate for different scanner coordinate systems and ensure reasonable overlap of common regions between two images. Fig. 6(a) and (b) show coronal slices from iCT and preCT images, respectively. Fig. 6(c) shows the overlay of these two images using the checkerboard pattern. In this checkerboard overlay, blocks from both the images are displayed alternately. The structural misalignment between iCT and preCT images is evident from mismatches at the boundaries of these blocks. Fig. 6(e) shows a coronal slice from the preCT image registered to the iCT image, and Fig. 6(f) shows the overlay of this image with the original iCT image. Better structural alignment after deformable registration is evident from improved matching at the boundaries of the blocks of the checkerboard overlay.

Similarly, Fig. 7 shows an example of deformable registration between a pair of iCT and PET images. This registration was also performed using the reported FPGA-based solution. The initial alignment between these two images was obtained as for the previous case. Fig. 7(a) and (b) show coronal slices from iCT and PET images, respectively. Fig. 7(c) shows the fusion of these two images. Fig. 7(d) shows the fusion of the registered PET image with the original iCT image. Improved alignment

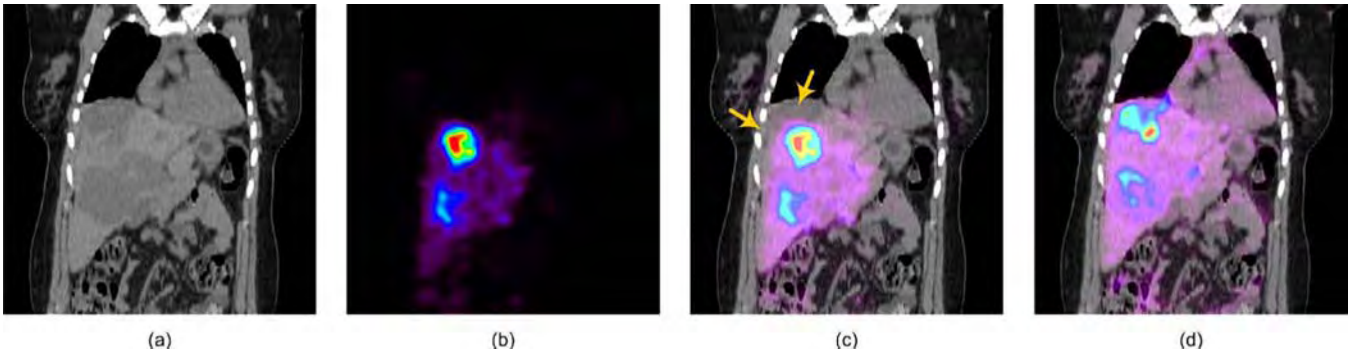


Fig. 7. Qualitative validation of deformable registration between iCT and PET images performed using the reported solution: (a) shows a coronal slice from an iCT image; (b) shows a corresponding coronal slice from a preprocedural PET image; (c) shows fusion of (a) and (b), indicating misalignment of liver (illustrated by arrows); and (d) shows fusion of (a) and registered PET, illustrating improved alignment of liver after deformable registration.

TABLE III
TARGET REGISTRATION ERROR (TRE) AFTER DEFORMABLE REGISTRATION

Image pair used for registration	Average TRE (mean ± standard deviation, mm)	
	Software implementation	FPGA-based implementation
iCT-preCT	3.63 ± 2.39	3.87 ± 2.52
iCT-PET	6.71 ± 3.02	6.92 ± 3.44

after deformable registration is evident from better matching of structures in the fusion image.

As described in Section III-B, quantitative validation of registration is performed by evaluating the correspondence of several anatomical landmarks. The TRE at these landmarks is then used as a metric to compare the registration accuracy of the high-speed FPGA-based implementation against the software implementation. For this purpose, all ten image pairs were registered using software implementation as well as the reported solution. The software registration was performed using a C++ implementation running on a modern workstation (Intel Xeon 3.6 GHz processor with 2-GB RAM). Table III reports the distribution of the average TRE at all landmarks for iCT-preCT and iCT-PET image pairs using both these solutions. Average TREs obtained for iCT-preCT and iCT-PET registration using the hardware implementation were 3.87 and 6.92 mm, respectively. Moreover, similar TRE values for software and hardware implementations indicate their comparable registration accuracy.

To ensure the reproducibility of the deformable registration results provided by the reported solution, we jittered the initial alignment between the image pairs by randomly generated values (within the range of ±20 voxels for translations and ±10° for rotations). The average TRE for all these attempts was similar to the reported TRE. This indicates the robustness of the reported FPGA-based implementation and its relative independence from the initial seed alignment.

V. DISCUSSION

Minimally invasive IGIs are time and cost efficient, lead to faster patient recovery, and as a result are becoming increasingly popular. For certain clinical procedures, such as liver biopsies, and cryo- or radiofrequency ablation (especially in the context of malignant masses in the liver), they have become the standard of care [40]–[42]. Multislice CT, with its continually improving coverage, imaging speed, and resolution has emerged as the primary choice of intraprocedural imaging modality during IGIs, compared with magnetic resonance imaging (which remains relatively slow) and 3-D ultrasound (which suffers from suboptimal image quality). Although these intraprocedural imaging modalities may provide the most current spatial information, they can suffer from poor target delineation resulting from lack of functional and/or contrast information during the procedure. Accurate, robust, and high-speed deformable registration will enable improved intraprocedural target delineation during IGIs through registration with preprocedural images. In this article, we presented an FPGA-based high-speed implementation of a deformable registration algorithm specially geared toward improving target delineation during CT-guided procedures.

Earlier reported techniques aimed at achieving this goal have primarily employed fiducial-based, mechanical alignment-based, segmentation-based, and/or intensity-based rigid alignment techniques [43]–[47]. These techniques are either not automatic or not retrospective and, more important, employ rigid-body approximation. This assumption may affect the accuracy and precision of the image-guided procedures when the underlying deformation is nonrigid, thus limiting their scope. The reported solution, in contrast, is fully automatic, completely retrospective, supports multimodality registration, and is capable of recovering nonrigid misalignment. In addition, the reported solution features high-speed execution of deformable registration through FPGA-accelerated implementation of MI calculation.

The majority of earlier reported attempts to accelerate intensity-based deformable registration have primarily employed a multiprocessor approach. Ourselin *et al.* [28] reported a parallel implementation of affine registration using a ten-processor cluster that provided a five-fold speedup. Similarly, Ino *et al.* [27] have reported a 10-min implementation of MI-based deformable registration using a 128-processor cluster. Another acceleration approach has been to use supercomputers, which offer a high degree of parallelism. Warfield *et al.* [30] performed deformable registration on a Sun supercomputer in 15 s. However, interactive segmentation of the brain surface in the intraoperative MR image took several minutes. Moreover, this implementation was specific to brain MR images because of high surface correspondence. Rohlfing *et al.* [29] have reported a speedup of 50 for a splines-based deformable registration algorithm using a 64-processor shared-memory supercomputer (SGI Origin 3800). Although these solutions delivered high performance by virtue of parallelization, the speedup achieved per processor was less than unity. Moreover, these solutions may not be cost effective and, because of their size, are unlikely to be suitable for clinical deployment. Other reported solutions based on graphics hardware [48]–[50] show how this promising platform can be utilized for certain image registration techniques. Although these solutions can be compact and low-cost, they cannot deliver the high-performance MI calculations required by IGI applications. In comparison, the reported solution is compact, low-cost, and offers a speedup of around 30 using a single computing element. Our group has earlier reported an FPGA-based design (FAIR-II) for accelerated calculation of MI [36]. Although this architecture offers more than a magnitude of speedup over software implementation, it is not suitable for volume subdivision-based image registration in its current form. The partitioning scheme employed in FAIR-II will result in prohibitively high MH memory requirement, especially since $\mathrm{MH}_{\mathrm{Rest}}$ must be calculated for every subvolume. Moreover, the reported design offers a voxel processing rate (100 MHz) twice that offered by FAIR-II (50 MHz).

Table I compares the time required for calculation of MI for a subvolume at every level of subdivision using a software implementation against that achieved using the FPGA-based implementation. The reported solution offers a speedup of about 40 for calculating MI up to subvolumes of size 64^3; whereas for smaller subvolumes the speedup achieved is around 20. This drop in achieved speedup can be explained by taking into account overheads incurred during computation of MI. Calculating MI requires accumulation of MH which, in turn, requires initial clearing of the MH memory. Because the reported implementation employs an MH with size 128 × 128 (to support 7-bit images), the process of clearing MH, which involves writing 0s to all MH entries, can consume more than 16,000 clock cycles. For smaller subvolumes, the time required to clear MH memory becomes comparable to or larger than that required to process a subvolume (images are processed at the rate of approximately one voxel per clock cycle). In addition, the communication time between the host and the MI calculator, required for exchanging the transformation matrix and the calculated MI value, becomes comparable to the computation time. These two factors limit the net speedup achieved for smaller subvolumes. As indicated by Table II, the reported solution provides a speedup of around 30 for deformable registration using both image modality pairs and achieves an execution time of around 6 min. This speedup is a direct outcome of acceleration of MI calculation using the presented architecture.

As described earlier, we evaluated the registration accuracy of the reported solution using five iCT-preCT image pairs and five iCT-PET image pairs. Table III summarizes the results of this accuracy analysis. The average TRE after deformable registration is on the order of a few voxels of the image with lower resolution (which controls the accuracy of intensity-based registration). In addition, the difference between the average TRE achieved by the high-speed FPGA implementation and a corresponding software implementation is subvoxel and is less than 0.25 mm.

The reported solution is capable of achieving deformable registration between a pair of images with size 256 × 256 × 256 in about 6 minutes, while providing accuracy comparable to a software implementation. This is a significant first step toward enabling integration of deformable registration in IGI workflow. Further acceleration of the aforementioned registration algorithm to satisfy the interactive requirement of IGIs can be achieved through additional strategies. The current architecture uses the same memory module to store both the RI and FI. Storing these images in two separate memory modules will allow their independent parallel access. This will eliminate the need to prefetch the RI voxels and thus provide speedup. In addition, using high-speed static random access memory (SRAM) modules for storing the randomly accessed FI is likely to provide further speedup by providing faster access to the FI with minimal latencies. Second, as showed by Studholme *et al.* [37], varying MH size between 32 × 32 and 256 × 256 does not significantly affect the accuracy of MI-based registration. Based on this observation, the size of the MH within the FPGA-based implementation can be adaptively reduced with every level of subdivision. This will reduce the overhead of clearing the MH for smaller subvolumes, thereby lending additional speedup. Finally, as described earlier, the algorithm optimizes the individual subvolumes at a given level of subdivision sequentially, but independently of each other. Thus, using multiple FPGA modules in parallel it is possible to simultaneously optimize these subvolumes. This multi-FPGA implementation will likely provide near-linear speedup. All these strategies in combination can further reduce the execution time of deformable image registration.

To summarize, the current work presents a novel FPGA-based architecture for high-speed implementation of MI-based deformable image registration. The reported architecture achieves a speedup of about 30 for both iCT-preCT and preCT-PET image pairs and reduces the time for deformable registration from hours to only a few minutes. The robustness, accuracy, and speed offered by the reported solution in conjunction with its compact implementation make it ideally suited for clinical deployment. Although the reported solution was developed and validated in the context of CT-guided interventions, the underlying deformable registration algorithm supports multimodal image registration. Thus, the reported solution can likely meet the deformable registration needs of a multitude of diagnostic and interventional applications.

VI. Conclusion

We have presented an FPGA-based architecture for high-speed implementation of MI-based deformable registration. This implementation enables improved target delineation during CT-guided interventions through deformable registration with preprocedural contrast-enhanced CT and PET images. The reported solution reduces the execution time of deformable registration from hours to only a few minutes, while providing comparable registration accuracy. The speed and accuracy offered by this solution, along with its compact implementation, make it suitable for integration into IGI workflow.

Minimally invasive IGIs are efficient, lead to faster recovery, and as a result are becoming increasingly popular. Accurate, robust, and real-time deformable image registration between intra- and preprocedural images is an unmet need, critical to the success of image-guided procedures. The work presented in this article is an important step toward meeting this goal. With further algorithmic and hardware improvements, this approach has the potential to elevate the precision of current procedures and expand the scope of IGI to moving and deformable organs.

Acknowledgment

The authors would like to thank Dr. V. Walimbe and Dr. N. Knight for their help in editing and refining this manuscript. The authors also thank the journal's reviewers for their feedback and suggestions in improving this manuscript.

References

[1] G. Antoch, J. F. Debatin, J. Stattaus, H. Kuehl, and F. M. Vogt, "Value of CT volume imaging for optimal placement of radiofrequency ablation probes in liver lesions," *J. Vasc. Intervent. Radiol.*, vol. 13, pp. 1155–1161, 2002.

[2] D. E. Dupuy and S. N. Goldberg, "Image-guided radiofrequency tumor ablation: Challenges and opportunities—Part II," *J. Vasc. Intervent. Radiol.*, vol. 12, pp. 1135–1148, 2001.

[3] J. R. Haaga, "Interventional CT: 30 years' experience," *Eur. Radiol.*, vol. 15, pp. d116–d120, 2005.

[4] E. K. Lang, F. Richter, L. Myers, R. A. Watson, R. J. Macchia, B. Gayle, and R. Thomas, "CT-guided biopsy of indeterminate renal cystic masses (Bosniak 3 and 2F): Accuracy and impact on clinical management," *Eur. Radiol.*, vol. 12, pp. 2518–2524, 2002.

[5] S. N. Goldberg and D. E. Dupuy, "Image-guided radiofrequency tumor ablation: Challenges and opportunities—Part I," *J. Vasc. Intervent. Radiol.*, vol. 12, pp. 1021–1032, 2001.

[6] K. Woertler, W. Winkelmann, W. Heindel, T. Vestring, F. Boettner, and N. Lindner, "Osteoid osteoma: CT-guided percutaneous radiofrequency ablation and follow-up in 47 patients," *J. Vasc. Intervent. Radiol.*, vol. 12, pp. 717–722, 2001.

[7] M. Das, F. Sauer, U. J. Schoepf, A. Khamene, K. Sebastian, S. Schaller, R. Kikinis, E. vanSonnenberg, and S. G. Silverman, "Augmented reality visualization for CT-guided interventions: System description, feasibility, and initial evaluation in an abdominal phantom," *Radiology*, vol. 240, pp. 230–235, 2006.

[8] C. Fujioka, J. Horiguchi, M. Ishifuro, H. Kakizawa, M. Kiguchi, N. Matsuura, M. Hieda, T. Tachikake, F. Alam, T. Furukawa, and K. Ito, "A feasibility study: Evaluation of radiofrequency ablation therapy to hepatocellular carcinoma using image registration of preoperative and postoperative CT," *Acad. Radiol.*, vol. 13, pp. 986–994, 2006.

[9] D. E. Heron, R. S. Andrade, and R. P. Smith, "Advances in image-guided radiation therapy-the role of PET-CT," *Med. Dosimetry*, vol. 31, pp. 3–11, 2006.

[10] P. Veit, C. Kuehle, T. Beyer, H. Kuehl, A. Bockisch, and G. Antoch, "Accuracy of combined PET/CT in image-guided interventions of liver lesions: An ex-vivo study," *World J. Gastroenterol.*, vol. 12, pp. 2388–2393, 2006.

[11] V. Vilgrain, "Tumour detection in the liver: Role of multidetector-row CT," *Eur. Radiol.*, vol. 15, pp. d85–d88, 2005.

[12] J. T. Yap, J. P. J. Carney, N. C. Hall, and D. W. Townsend, "Image-guided cancer therapy using PET/CT," *Cancer J.*, vol. 10, pp. 221–233, 2004.

[13] S. B. Solomon, "Incorporating CT, MR imaging, and positron emission tomography into minimally invasive therapies," *J. Vasc. Intervent. Radiol.*, vol. 16, pp. 445–447, 2005.

[14] D. J. Hawkes, J. McClelland, C. Chan, D. L. G. Hill, K. Rhode, G. P. Penney, D. Barratt, P. J. Edwards, and J. M. Blackall, "Tissue deformation and shape models in image-guided interventions: A discussion paper," *Med. Image Anal.*, vol. 9, pp. 163–175, 2005.

[15] T. Carter, M. Sermesant, D. Cash, D. Barratt, C Tanner, and D Hawkes, "Application of soft tissue modelling to image-guided surgery," *Med. Eng. Phys.*, vol. 27, pp. 893–909, 2005.

[16] F. Maes, A. Collignon, D. Vandermeulen, G. Marchal, and P. Suetens, "Multimodality image registration by maximization of mutual information," *IEEE Trans. Med. Imag.*, vol. 16, no. 2, pp. 187–198, Apr. 1997.

[17] V. Walimbe and R. Shekhar, "Automatic elastic image registration by interpolation of 3-D rotations and translations from discrete rigid-body transformations," *Med. Image Anal.*, vol. 10, pp. 899–914, 2006.

[18] V. Walimbe, V. Zagrodsky, S. Raja, B. Bybel, M. Kanvinde, and R. Shekhar, "Elastic registration of three-dimensional whole body CT and PET images by quaternion-based interpolation of multiple piecewise linear rigid-body registrations," *SPIE Med. Imag.: Image Process.*, pp. 119–128, 2004.

[19] R. Shekhar, V. Walimbe, S. Raja, V. Zagrodsky, M. Kanvinde, G. Wu, and B. Bybel, "Automated 3-dimensional elastic registration of whole-body PET and CT from separate or combined scanners," *J. Nucl. Med.*, vol. 46, pp. 1488–1496, 2005.

[20] D. Mattes, D. R. Haynor, H. Vesselle, T. K. Lewellen, and W. Eubank, "PET-CT image registration in the chest using free-form deformations," *IEEE Trans. Med. Imag.*, vol. 22, no. 1, pp. 120–128, Jan. 2003.

[21] O. Dandekar, K. Siddiqui, V. Walimbe, and R. Shekhar, "Image registration accuracy with low-dose CT: How low can we go?," in *Proc. 3rd IEEE Int. Symp. Biomed. Imag.: Nano to Macro*, 2006, pp. 502–505.

[22] P. Lei, O. Dandekar, F. Mahmoud, D. Widlus, P. Malloy, and P. M. R. Shekhar, "PET guidance for liver radiofrequency ablation: An evaluation," in *Proc. Med. Imag.: Vis. Image-Guided Procedures*, San Diego, CA, 2007, p. 650918.

[23] R. Shekhar, P. Lei, C. R. Castro-Pareja, W. L. Plishker, and W. D. D'Souza, "Automatic segmentation of phase-correlated CT scans through nonrigid image registration using geometrically regularized free-form deformation," *Med. Phys.*, vol. 34, pp. 3054–3066, 2007.

[24] R. Shekhar, P. Lei, V. Walimbe, C. Yu, and W. D'Souza, "A novel volume subdivision-based algorithm to register respiration phase-correlated lung CT images," *Med. Phys.*, vol. 32, pp. 2083–2083, 2005.

[25] V. Walimbe, O. Dandekar, F. Mahmoud, and R. Shekhar, "Automated 3-D elastic registration for improving tumor localization in whole-body PET-CT from combined scanner," in *Proc. IEEE 28th Annu. Int. Conf. EMBSE*, 2006, pp. 2799–2802.

[26] J. Wu, O. Dandekar, V. Walimbe, W. D'Souza, and R. Shekhar, "Automatic prostate localization using elastic registration of planning CT and daily 3-D ultrasound images," in *Proc. Med. Imag.: Vis. Image-Guided Procedures*, San Diego, CA, 2007, p. 650913.

[27] F. Ino, K. Ooyama, and H. Kenichi, "A data distributed parallel algorithm for nonrigid image registration," *Parallel Comput.*, vol. 31, pp. 19–43, 2005.

[28] S. Ourselin, R. Stefanescu, and X. Pennec, "Robust registration of multi-modal images: Towards real-time clinical applications," in *Proc. 5th Int. Conf. MICCAI—Part II*, 2002, pp. 140–147.

[29] T. Rohlfing and C. R. Maurer, "Nonrigid image registration in shared-memory multiprocessor environments with application to brains, breasts, and bees," *IEEE Trans. Inf. Technol. Biomed.*, vol. 7, no. 1, pp. 16–25, Mar. 2003.

[30] S. K. Warfield, F. Talos, A. Tei, A. Bharatha, A. Nabavi, M. Ferrant, P. M. Black, F. A. Jolesz, and R. Kikinis, "Real-time registration of volumetric brain MRI by biomechanical simulation of deformation during image guided neurosurgery," *Comput. Visual. Sci.*, vol. 5, pp. 3–11, 2002.

[31] C. R. Castro-Pareja, J. M. Jagadeesh, and R. Shekhar, "FAIR: A hardware architecture for real-time 3-D image registration," *IEEE Trans. Inf. Technol. Biomed.*, vol. 7, no. 4, pp. 426–434, Dec. 2003.

[32] M. Doggett and M. Meissner, "A memory addressing and access design for real time volume rendering," in *Proc. IEEE ISCAS*, 1999, pp. 344–347.

[33] H. Hassler and N. Takagi, "Function evaluation by table look-up and addition," in *Proc. IEEE Symp. Comput. Arithmet.*, 1995, pp. 10–16.

[34] D. M. Mandelbaum and S. G. Mandelbaum, "A fast, efficient parallel-acting method of generating functions defined by power series, including logarithm, exponential, and sine, cosine," *IEEE Trans. Parallel Distributed Syst.*, vol. 7, no. 1, pp. 33–45, Jan. 1996.

[35] S. L. SanGregory, C. Brothers, D. Gallagher, and R. Siferd, "A fast, low-power logarithm approximation with CMOS VLSI implementation," in *Proc. IEEE 42nd Midwest Symp. Circuits Syst.*, 2000, pp. 388–391.

[36] C. R. Castro-Pareja and R. Shekhar, "Hardware acceleration of mutual information-based 3-D image registration," *J. Imag. Sci. Technol.*, vol. 49, pp. 105–113, 2005.

[37] C. Studholme, D. L. G. Hill, and D. J. Hawkes, "Overlap invariant entropy measure of 3-D medical image alignment," *Pattern Recognit.*, vol. 32, pp. 71–86, 1999.

[38] C. R. Castro-Pareja and R. Shekhar, "Adaptive reduction of intensity levels in 3-D images for mutual information-based registration," in *Proc. Med. Imag. 2005: Image Process.*, San Diego, CA, 2005, pp. 1201–1212.

[39] O. Dandekar, C. Castro-Pareja, and R. Shekhar, "FPGA-based real-time 3-D image preprocessing for image-guided medical interventions," *J. Real-Time Image Process.*, vol. 1, pp. 285–301, 2007.

[40] A. J. Bilchik, D. M. Rose, D. P. Allegra, P. J. Bostick, E. Hsueh, and D. L. Morton, "Radiofrequency ablation: A minimally invasive technique with multiple applications," *Cancer J. Scient. Amer.*, vol. 5, pp. 356–361, 1999.

[41] J. H. Morgan, G. M. Royer, P. Hackett, T. C. Gamblin, B. L. McCampbell, A. Conforti, and P. S. Dale, "Radio-frequency ablation of large, nonresectable hepatic tumors," *Am. Surg.*, vol. 70, pp. 1035–1038, 2004.

[42] D. M. Rose, D. P. Allegra, P. J. Bostick, L. J. Foshag, and A. J. Bilchik, "Radiofrequency ablation: A novel primary and adjunctive ablative technique for hepatic malignancies," *Am. Surg.*, vol. 65, pp. 1009–1014, 1999.

[43] C. Kremser, C. Plangger, R. Bosecke, A. Pallua, F. Aichner, and S. R. Felber, "Image registration of MR and CT images using a frameless fiducial marker system," *Magn. Reson. Imag.*, vol. 15, pp. 579–585, 1997.

[44] G. P. Penney, J. M. Blackall, M. S. Hamady, T. Sabharwal, A. Adam, and D. J. Hawkes, "Registration of freehand 3-D ultrasound and magnetic resonance liver images," *Med. Image Anal.*, vol. 8, pp. 81–91, 2004.

[45] R. C. Susil, J. H. Anderson, and R. H. Taylor, "A single image registration method for CT guided interventions," in *Proc. Med. Image Comput. Computer-Assisted Intervention (MICCAI)*, Berlin, Germany, 1999, pp. 798–808.

[46] J. Weese, G. P. Penney, P. Desmedt, T. M. Buzug, D. L. G. Hill, and D. J. Hawkes, "Voxel-based 2-D/3-D registration of fluoroscopy images and CT scans for image-guided surgery," *IEEE Trans. Inf. Technol. Biomed.*, vol. 1, no. 4, pp. 284–293, Dec. 1997.

[47] B. J. Wood, H. Zhang, A. Durrani, N. Glossop, S. Ranjan, D. Lindisch, E. Levy, F. Banovac, J. Borgert, S. Krueger, J. Kruecker, A. Viswanathan, and K. Cleary, "Navigation with electromagnetic tracking for interventional radiology procedures: A feasibility study," *J. Vasc. Intervent. Radiol.*, vol. 16, pp. 493–505, 2005.

[48] R. Strzodka, M. Droske, and M. Rumpf, "Fast image registration in DX9 graphics hardware," *J. Med. Informatics Technol.*, vol. 6, pp. 43–49, 2003.

[49] C. Vetter, C. Guetter, C. Xu, and R. Westermann, "Non-rigid multimodal registration on the GPU," in *Proc. Med. Imag. 2007: Image Process.*, San Diego, CA, 2007, p. 651228.

[50] A. Köhn, J. Drexl, F. Ritter, M. König, and H. Peitgen, "GPU accelerated image registration in two and three dimensions," *Bildverarbeitung Für die Medizin 2006*, vol. 3, pp. 261–265, 2006.

Omkar Dandekar (M'02) received the B.E. degree in biomedical engineering from the University of Mumbai, Mumbai, India, in 2000, and the M.S. degree in electrical engineering from Ohio State University, Columbus, in 2004. He is currently working toward the Ph.D. degree in the Department of Electrical and Computer Engineering, University of Maryland, College Park.

He has worked as a graduate research assistant at the Cleveland Clinic Foundation. His primary interests include medical imaging, digital VLSI design, and hardware acceleration of image processing algorithms. Currently, his research work is focused on real-time 3-D imaging and advanced image processing and analysis for image-guided interventions.

Raj Shekhar (M'94) received the B.Tech. degree in electrical engineering from the Indian Institute of Technology, Kanpur, India, in 1989, the M.S. degree in bioengineering from the Arizona State University, Tempe, in 1991, and the Ph.D. degree in biomedical engineering from the Ohio State University, Columbus, in 1997.

He is an Assistant Professor of Diagnostic Radiology, Bioengineering, and Electrical and Computer Engineering at the University of Maryland, Baltimore and College Park. He previously served as a Staff Scientist at the Cleveland Clinic and as a Senior Engineer at Picker International (now Philips Medical Systems). His research interests include medical image processing, real-time computing, 3-D ultrasound, and image-guided interventions. He has authored more than 60 scientific papers, including more than 25 peer-reviewed articles. He also holds four U.S. patents.

Appendix F:

A statistical approach to high-quality CT reconstruction at low radiation doses for real-time guidance and navigation

Avanti Shetye[a,b] and Raj Shekhar[*,a,b]

[a]Department of Electrical and Computer Engineering, University of Maryland, College Park, MD, USA 20740
[b]Department of Diagnostic Radiology, University of Maryland, Baltimore, MD, USA 21201

ABSTRACT

The advent of 64-slice computed tomography (CT) with high-speed scanning makes CT a highly attractive and powerful tool for navigating image-guided procedures. For interactive navigation, scanning will need to be performed over extended time periods or even continuously. However, continuous CT is likely to expose the patient and the physician to potentially unsafe levels of radiation. Before CT can be used appropriately for navigational purposes, the dose problem must be solved. Simple dose reduction is not adequate, because it degrades image quality. This problem can be overcome if the traditional filtered back-projection (FBP) reconstruction is replaced with the maximum likelihood expectation maximization (MLEM) approach. MLEM is more accurate in that it incorporates Poisson statistics of the noisy projection data, especially at low doses. Our study shows that MLEM reconstruction is able to reduce x-ray dose from 200 to 11 mAs (the lowest dose-simulator setting in the present study) without significant image degradation. Taking advantage of modern CT scanners and specialized hardware, it may be possible to perform continuous CT scanning at acceptable radiation doses for intraoperative visualization and navigation.

Keywords: CT, image reconstruction, image-guided procedure, intraoperative guidance and navigation, metal artifact reduction

1. INTRODUCTION

The method of choice for many surgical procedures has shifted from traditional open surgery to the use of less invasive means, a transition facilitated by the introduction of minimally invasive techniques more than a decade ago. Such procedures are often performed through about 3 or 4 small skin ports (keyhole-size holes) instead of the 6- to 8-inch incisions required for traditional surgery[1]. The results are reduced trauma to the body, shorter recovery times and lower costs. However, the utility of such procedures is limited without a clear representation of the anatomy undergoing procedure. The ability of the clinician will be greatly enhanced if a three-dimensional (3D) visualization of this anatomy is available to guide such procedures.

Computerized tomography (CT), a widely used diagnostic technique, is known to provide a highly accurate volumetric representation of the anatomy, with good contrast resolution. A CT scanner can create instantaneous 3D representations of the internal anatomy with good contrast resolution. The speed of image reconstruction gives CT an edge over other imaging modalities in terms of continuous visualization of and navigation through structures. Some minimally invasive procedures utilize this benefit by acquiring a preoperative CT scan for guidance. This approach is limited because it does not provide updated information on intraoperative anatomic deformations. A continuous CT guided approach can represent intraoperative anatomy accurately, but such scanning is practical only if radiation is reduced to a minimal level. Commercially available CT scanners employ a filtered backprojection (FBP) technique for image reconstruction. Although useful in many imaging applications, the FBP technique does not allow dose reduction without significantly degrading image quality. Continuous CT with FBP reconstruction, then, would expose patient and practitioner to elevated dose levels.

*rshekhar@umm.edu, phone (410)706-8714, fax (410)706-8724

Medical Imaging 2007: Physics of Medical Imaging, edited by iang sieh, Michael . lynn,
Proc. of SPIE Vol. 6510, 65105 , (2007) · 1605-7422/07/$18 · doi: 10.1117/12.710107

FBP also causes streak artifacts when metal is in the field of view, for example, during surgery. In this study, we propose the replacement of FBP with a statistical approach using maximum likelihood expectation maximization (MLEM) for image reconstruction. The motivation behind this study is to utilize the benefit of 3D visualization achieved through CT, but at a greatly reduced radiation dose without compromising image quality. Iterative techniques using maximum likelihood are proven to replicate Poisson statistics for positron emission tomography, single photon emission computed tomography and CT[2-7]. Although statistical reconstruction is computationally expensive, the suboptimal FBP approach is certainly not acceptable for reconstructing noisy projection data. **Our purpose is to show the achievability of dose reduction for continuous CT scanning using the MLEM algorithm**.

FBP, the conventional approach to CT reconstruction, uses the Fourier slice theorem to arrive at a closed form deterministic solution to finding attenuation coefficients[8]. The underlying assumption behind this theorem is that each projection represents an independent measurement of the object[8]. The advantage of FBP is that the process of reconstruction can be started as soon as the first projection has been measured, speeding up the process and reducing the requirements for storage. FBP reconstruction produces high-quality images at high radiation doses. However, the image quality begins to deteriorate as the x-ray dose is reduced. Dose reduction is a crucial requirement for the application of CT in interventional procedures, where patients and practitioners will be exposed to continuous radiation over the duration of the surgery.

The process of photon-generation in an x-ray tube can be approximated using the Poisson distribution. Iterative techniques such as MLEM capture the stochastic variations in photon counts accurately (unlike the deterministic FBP approach) yielding more accurate reconstructions at much lower radiation doses. Maximum likelihood has been shown to have excellent theoretical properties that model the statistical nature of CT in a realistic manner[6]. The objective of this algorithm is to maximize the complete likelihood of the photons entering each pixel along the projection ray, given the number of photons detected by the detector at the projection, parameterized by the current estimate of pixel intensities. The new estimate of the pixel intensity can be approximated to a closed-form solution. The original MLEM algorithm is presented in brief in the next section.

2. METHODS AND MATERIALS

Our method has been developed based on the Lange and Carson[6] framework. The concept is described for parallel beam geometry and can be extended easily to fan beam geometry.

The number of photons detected by scanning air provides a fair approximation of the number of photons generated by the x-ray source. If W_i is the number of photons leaving the source, all W_i will be detected in the absence of an attenuating object. In the presence of an attenuating object, if Y_i is the number of photons detected, then by Beer's law, each photon leaving the source has an equal probability of reaching the detector. This probability is expressed as:

$$p_i = e^{-\sum_{j \in J_i} l_{ij} \mu_j} , \tag{1}$$

where l_{ij} is the length of intersection of the ith ray with the jth pixel, μ_j is the intensity of the jth pixel and J_i is the set of all pixels traversed by the ith ray.
Because Y_i follows a Poisson distribution, the entire log likelihood can be reduced to

$$\ln g(Y,\mu) = \sum_i \left\{ -W_i e^{-\sum_{j \in J_i} l_{ij}\mu_j} - Y_i \sum_{j \in J_i} l_{ij}\mu_j + Y_i \ln W_i - \ln Y_i! \right\} . \tag{2}$$

The strict concavity, which suggests the existence of a maximum of this likelihood, can be established by the non-negative definiteness of the matrix with elements

$$a_{ik} = \begin{cases} l_{ik} \ldots\ldots.k \in J_i \\ 0 \ldots\ldots\ldots k \notin J_i \end{cases} . \tag{3}$$

In the MLEM algorithm, a reconstruction grid of uniform intensity is used as the initial estimate. Iterating on the reconstruction grid, the log likelihood is maximized and the maximizing image estimate is used as an initial estimate for the next iteration. The closed form solution at the $(n+1)^{th}$ iteration is expressed as:

$$\mu_k^{\ n+1} = \frac{\sum_{i \in J_i} \left(M_{ik} - N_{ik} \right)}{\frac{1}{2} \sum_{i \in J} \left(M_{ik} + N_{ik} \right) l_{ik}} \quad , \tag{4}$$

where M_{ik} and N_{ik} are the expected number of photons entering and leaving pixel k and are determined through Beer's law (Eq. 1).

The reconstruction algorithm was applied to simulated data from a 512 x 512 digital Shepp-Logan phantom and to real projection data from an abdominal phantom representing real anatomy. The reconstruction quality was assessed using power signal-to-noise ratio (PSNR) for the digital phantom calculated as:

$$PSNR = 10 \log_{10} \frac{(2^{bitdepth} - 1)^2}{MSE(A,B)} \quad , \tag{5}$$

$$MSE(A,B) = \frac{\sum_{i=1}^{M} \sum_{j=1}^{N} \left(A_{ij} - B_{ij} \right)^2}{MN} \quad , \tag{6}$$

Where M x N represents the number of pixels in image A and B, A_{ij} represents the intensity of $(i,j)^{th}$ pixel of A and B_{ij} represents the intensity of $(i,j)^{th}$ pixel of B.

3. RESULTS

3.1 Digital Shepp-Logan phantom

A 512 x 512 digital Shepp-Logan phantom was generated in MATLAB and projection data was simulated using Beer's law (1). Expectation of noise in low-dose sinograms was estimated by fitting a Poisson distribution to the difference between sinograms of images obtained from the low-dose simulator at 200 mAs and at lower doses. This Poisson noise, with the assumption that sinogram noise follows Poisson distribution, was added to the simulated projections for the digital phantom to generate noisy data resembling low-dose (low tube current) projections. Reconstructions using MLEM algorithm yielded better results in terms of PSNR values with the original phantom as the reference image, than did corresponding reconstructions using FBP.

The digital phantom used in our study is shown in Figure 1. Reconstructions at 11mAs using FBP and MLEM are shown in Figure 2(a) for a visual comparison. To test the reproducibility of our results, reconstructions at 15mAs using the same 2 methods are shown in Figure 2(b). At each of these doses, MLEM led to higher contrast resolution, mimicking that of the original image. PSNRs for these two algorithms at a range of dose levels are summarized in Table 1. A quantitative comparison of the reconstruction qualities achieved through FBP and MLEM is delineated by means of a plot in Figure 3. The comparison shows that MLEM outperforms FBP at any given dose level. Note that a tube current setting of 11mAs is the lowest achievable dose on a Siemens dose simulator. [Courtesy: Baltimore Veteran Affairs Medical Center, MD]

3.2 Abdominal phantom

An abdominal phantom was scanned using a Philips Brilliance 40-slice CT scanner at 120 kV and tube current varying at random from 25 milli-ampere second (mAs) to 250 mAs. Axial scanning was done at 2-sec cycle time with standard resolution, 16 x 2.5 mm collimation and slice thickness of 5 mm. The number of bins per view was 672, with 1160 views evenly spanned on a circular orbit of 360^o. Raw unpreprocessed fan beam data were extracted using scanner software and altered to parallel beam data. Image quality (in terms of PSNR) of MLEM reconstruction degrades less precipitously than that of FBP as the dose level is reduced from 250 to 25 mAs. Figure 4 and Figure 5 provide visual assessments of FBP and MLEM reconstructions. A quantitative assessment is detailed in Table 2 and shown in Figure 6.

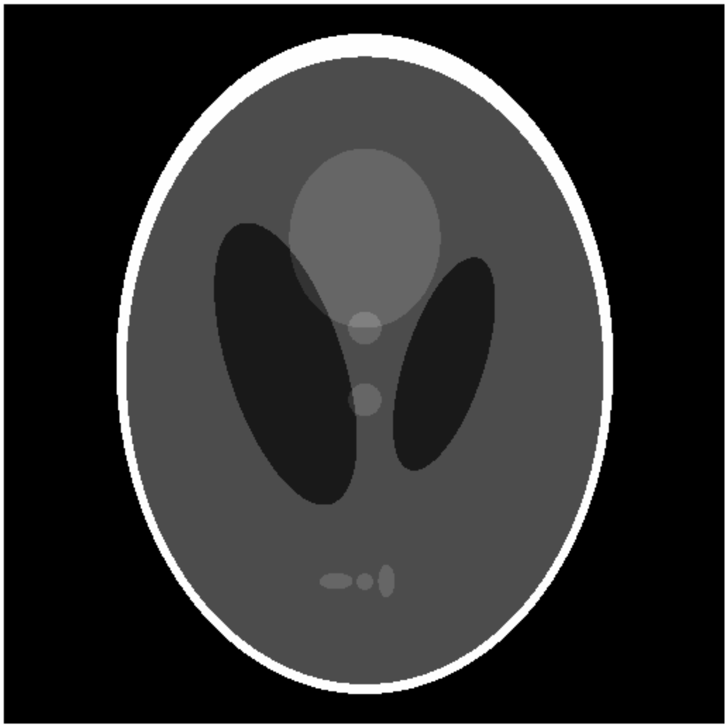
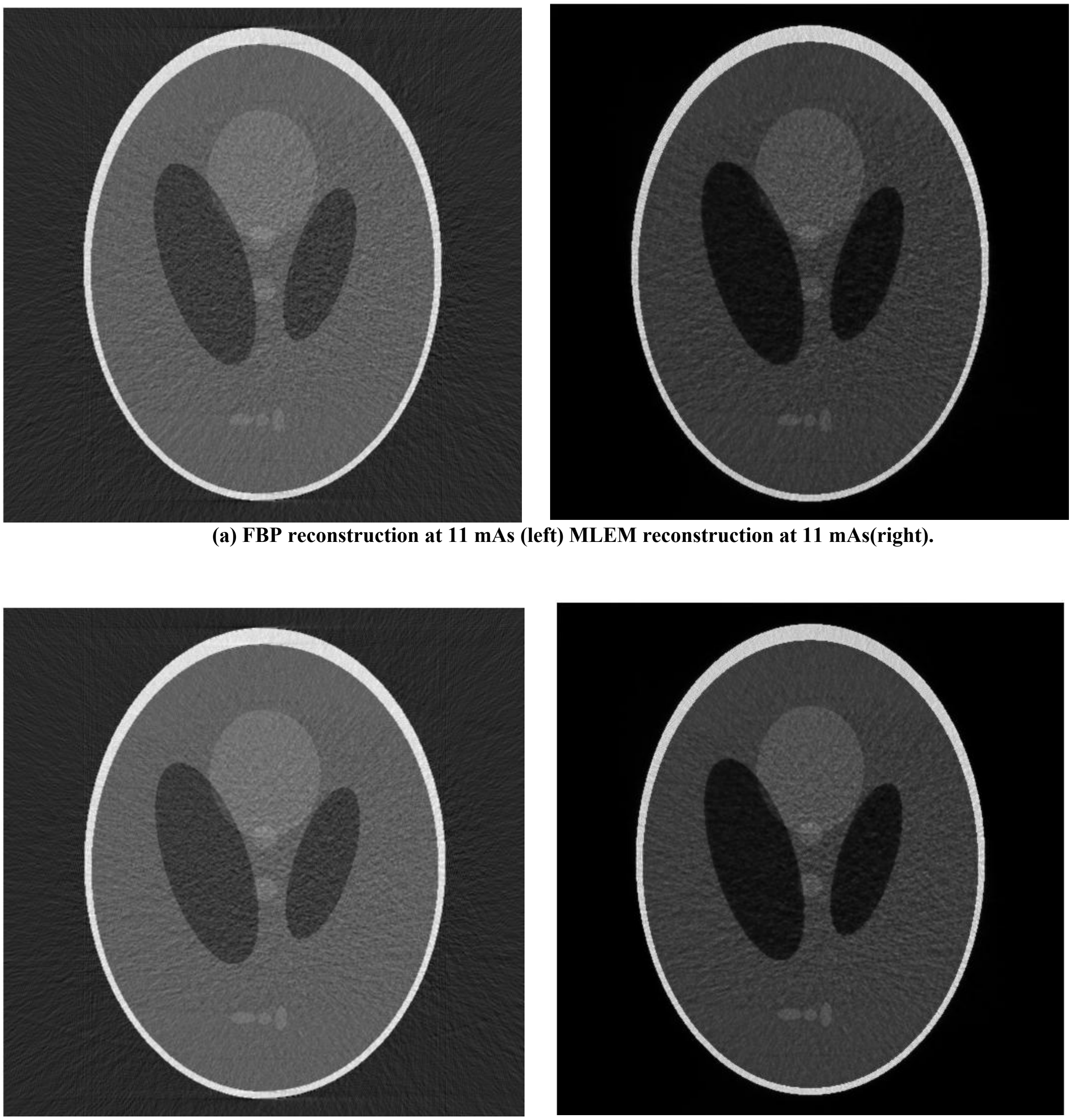


Figure 1: A 512 x 512 digital Shepp-Logan phantom

Table 1: PSNR comparison between FBP and MLEM for digital phantom

Dose in mAs	PSNR using FBP (dB)	PSNR using MLEM (dB)
11	**30.08**	**36.98**
15	**31.18**	**37.67**
20	**32.73**	**38.56**
25	**33.09**	**38.83**
30	**33.97**	**39.50**
40	**34.88**	**39.96**
50	**35.86**	**40.13**
70	**37.00**	**40.67**
85	**37.26**	**40.90**
100	**37.36**	**40.99**
150	**38.56**	**41.07**



(a) FBP reconstruction at 11 mAs (left) MLEM reconstruction at 11 mAs(right).

(b) FBP reconstruction at 15 mAs (left) MLEM reconstruction at 15 mAs (right).

Figure 2: Visual comparison of reconstruction quality.

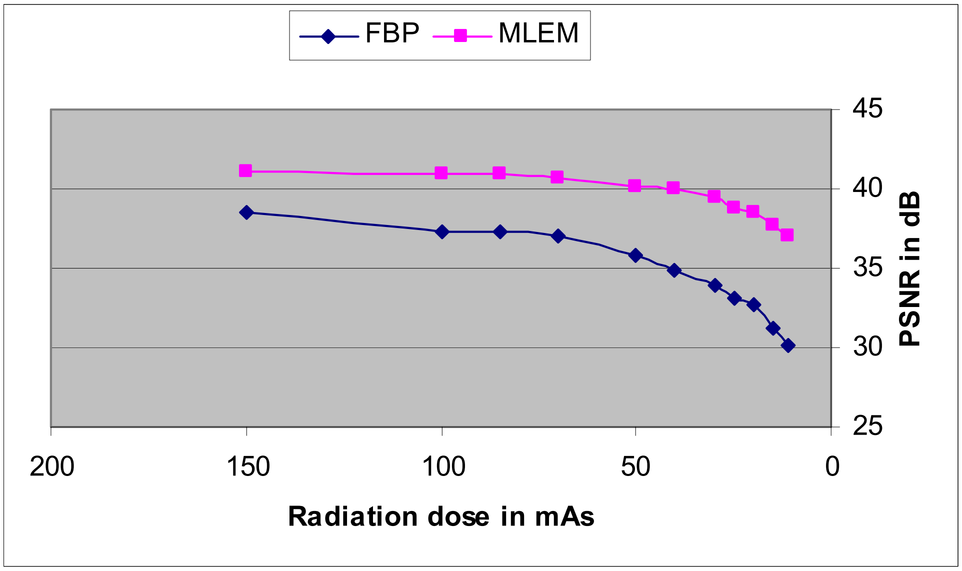


Figure 3: PSNR comparison between FBP and MLEM.

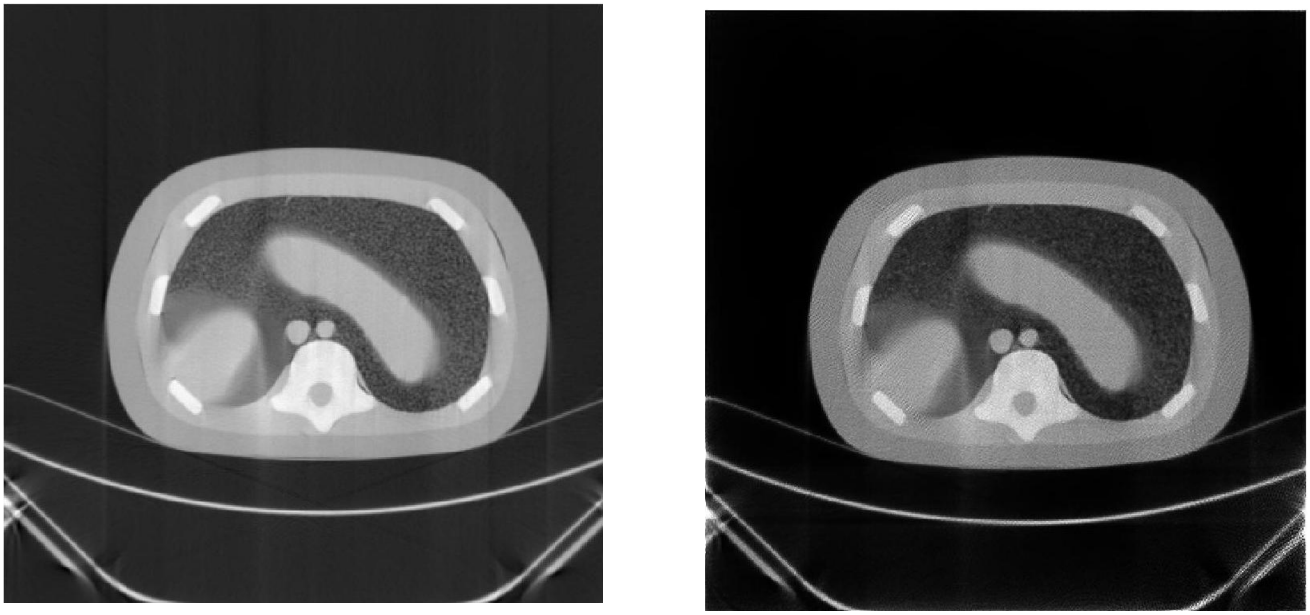


Figure 4: FBP reconstruction (left) and MLEM reconstruction (right) of abdominal phantom at 200 mAs.

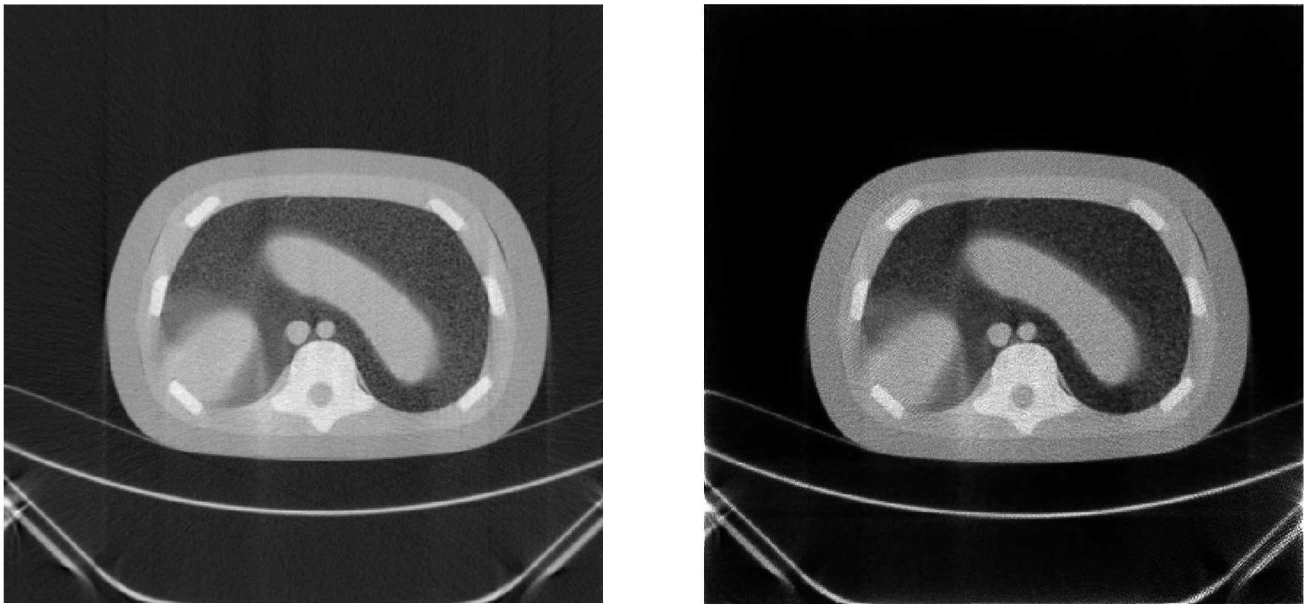


Figure 5: FBP reconstruction (left) and MLEM reconstruction (right) of abdominal phantom at 25 mAs.

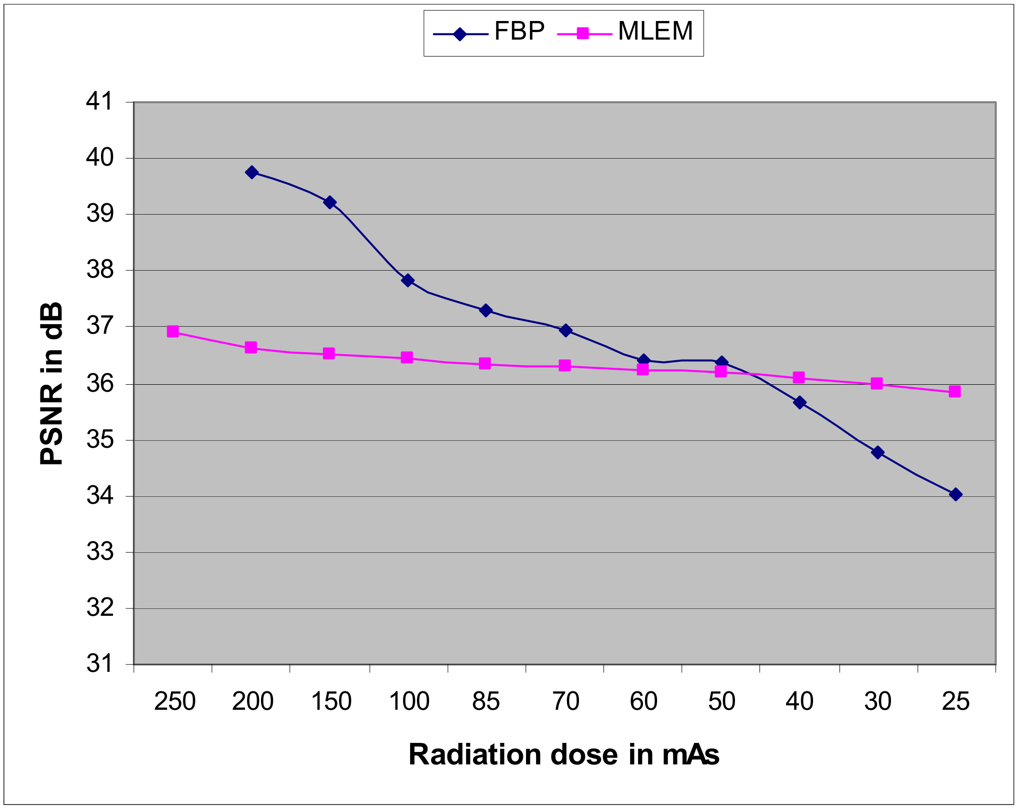


Figure 6: PSNR comparison between FBP and MLEM for abdominal phantom

Table 2: PSNR comparison between FBP and MLEM for abdominal phantom

Dose in mAs	PSNR using MLEM (dB)	PSNR using FBP (dB)
250	**36.90**	**Inf**
200	**36.63**	**39.75**
150	**36.51**	**39.20**
100	**36.46**	**37.85**
85	**36.33**	**37.31**
70	**36.30**	**36.93**
60	**36.23**	**36.40**
50	**36.19**	**36.37**
40	**36.08**	**35.65**
30	**35.97**	**34.79**
25	**35.85**	**34.02**

3.3. Metal artifact reduction

To demonstrate metal artifact reduction, a high-attenuation object was introduced in a 512 x 512 digital Shepp-Logan phantom at the pixel location (190, 295) shown in Figure 7. The projection data (Figure 8) simulated using Beer's law (Eq. 1) was reconstructed using the MLEM algorithm depicted in Figure 10.

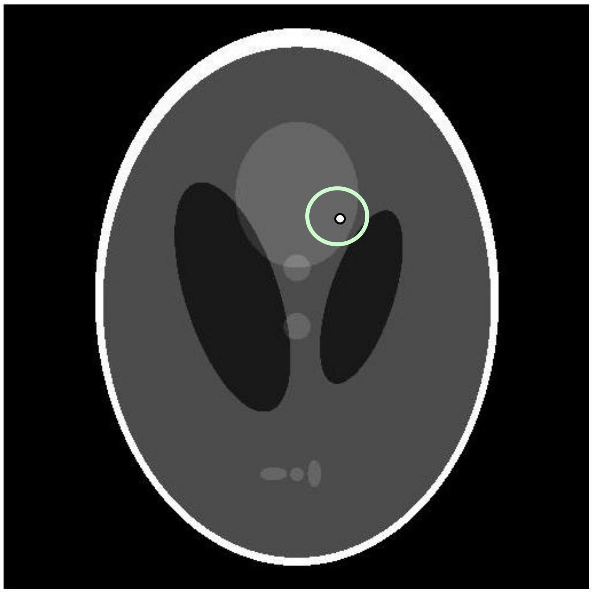


Figure 7: Digital Shepp-Logan phantom with high-attenuation pixel at location (190,295). (highlighted for clarity)

With the prior knowledge of the location of the high-attenuation object, the MLEM algorithm was able to accurately eliminate the FBP-algorithm generated streak artifacts[9,11] by disregarding projections passing through the pixels occupied by the metal to compute the likelihood (Figure 9). The approach becomes practical if a priori knowledge of the location of rigid metallic tools and their attenuation coefficients in the field of view of the scanner is available using specialized commercial tracking tools (such as those marketed by Polaris).

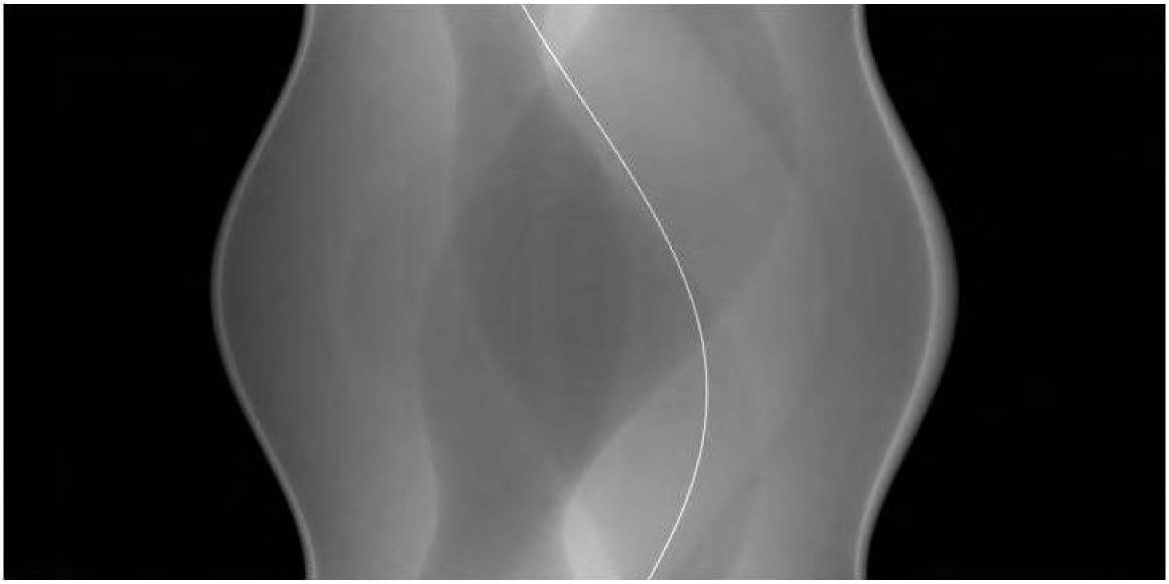


Figure 8: Parallel beam sinogram of the digital phantom in Figure 7 with a high-intensity pixel (number of projections in degrees against number of detectors).

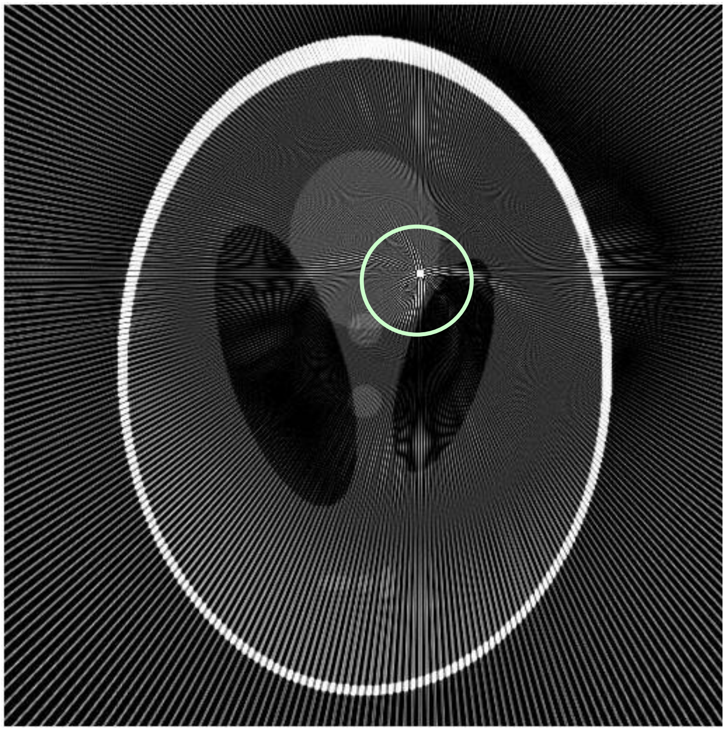


Figure 9: Streak artifacts from metal after reconstruction using FBP (highlighted for clarity).

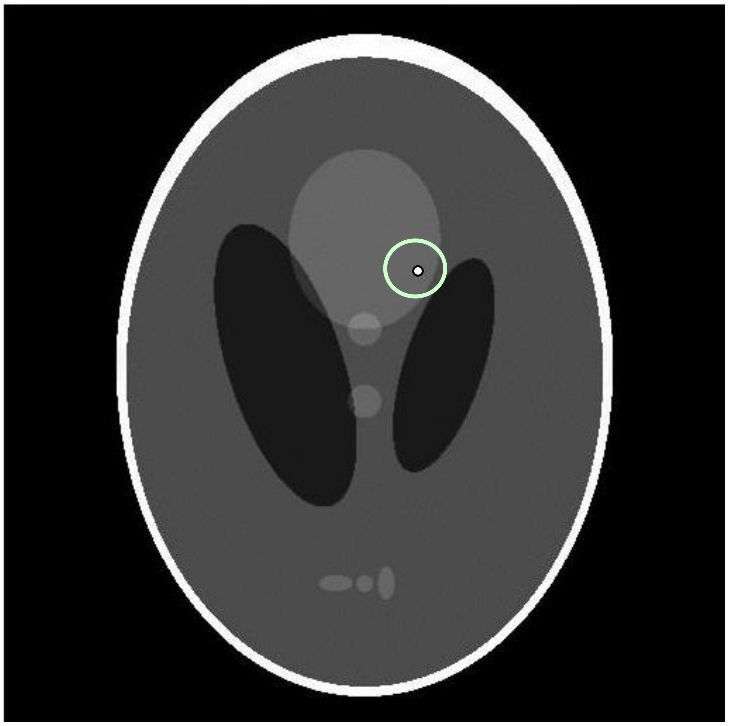


Figure 10: Metal artifact reduction using MLEM and tracking information.

4. DISCUSSION

We have demonstrated a reduction in x-ray radiation using the MLEM algorithm. The iterative MLEM algorithm incorporates the stochastic properties of x-ray photons while deriving a closed-form solution for attenuation coefficients. The image quality of MLEM at low dose was consistently better than that of the corresponding FBP reconstruction for the digital phantom. For the abdominal phantom, the image quality degraded more rapidly for FBP than for MLEM with the reduction of radiation dose. Although PSNR provides a good estimate of image quality for a digital phantom, it is not the best assessment measure for clinical images because of the absence of a standard reference for comparison. PSNR does not correlate strongly with subjective image quality ratings or observer task performance, limiting its utility in image quality assessment investigations[12]. Other assessment measures, such as the just noticeable difference[13] measure should be investigated to provide an accurate comparison between MLEM and FBP.

The integration of our algorithm in the clinical setting with the use of specialized tracking instruments and markers will successfully eliminate metal artifacts resulting from tools in the field of view. The use of such tracking instruments in a clinical setting was successfully tested in an animal experiment as part of our Operating Room of the Future research.

5. SUMMARY

Accurate and interactive navigation of image-guided procedures relies on high frame-rate intraoperative imaging and 3D visualization of the involved anatomy, such as that possible with 64-slice CT. Reduced dose will minimize the risks associated with prolonged radiation exposure. The achievement of dose reduction, as presented here, establishes the feasibility of an innovative continuous CT-guided visualization and navigation system. This study provides proof-of-concept evidence for dose reduction in two dimensions using an approach that can be extended easily to three dimensions.

ACKNOWLEDGEMENTS

This research is supported by US Department of Defense grant DAMD17-03-2-001.

REFERENCES

1. Cleaveland Clinic Foundation, “Minimally Invasive Cardiovascular and Thoracic Surgeries”, 20 Nov. 2006, <http://www.clevelandclinic.org/heartcenter/pub/guide/disease/mini_invasivehs.htm>
2. J. Browne and A. R. De Pierro, "A row-action alternative to the EM algorithm for maximizing likelihoods in emission tomography," *IEEE Trans. Med. Imaging* **15**, pp. 687-699, 1996.
3. B. De Man, J. Nuyts, P. Dupont, G. Marchal, and P. Suetens, "An iterative maximum-likelihood polychromatic algorithm for CT," *IEEE Trans. Med. Imaging* **20**, pp. 999-1008, 2001.
4. I. A. Elbakri and J. A. Fessler, "Statistical image reconstruction for polyenergetic x-ray computed tomography," *IEEE Trans. Med. Imaging* **21**, pp. 89-99, 2002.
5. J. A. Fessler and A. O. Hero, "Penalized maximum-likelihood image reconstruction using space-alternating generalized EM algorithms," *IEEE Trans. Image Process.* **4**, pp. 1417-1429, 1995.
6. K. Lange and R. Carson, "EM reconstruction algorithms for emission and transmission tomography," *Jour. Comput. Assist. Tomogr.* **8**, pp. 306-316, 1984.
7. L. A. Shepp and Y. Vardi, "Maximum likelihood reconstruction for emission tomography," *IEEE Trans. Med. Imaging* **1**, pp. 113-122, 1982.
8. A. C. Kak and M. Slaney, *Principles of Computerized Tomographic Imaging*. New York: IEEE press, NY, 1988.
9. P. J. La Rivière and D. Billmire, "Reduction of noise-induced streak artifacts in x-ray computed tomography through spline-based penalized-likelihood sinogram smoothing," *IEEE Trans. Med. Imaging* **24**, pp. 105-111, 2005.
10. P. J. La Rivière and X. Pan, "Nonparametric regression sinogram smoothing using a roughness-penalized Poission likelihood objective function," *IEEE Trans. Med. Imaging* **19**, pp. 773-786, 2000.
11. G. Wang, T. Frei, and M. W. Vannier, "Fast iterative algorithm for metal artifact reduction in x-ray CT," *Acad. Radiol.* **7**, pp. 607-614, 2000.
12. W. F. Good, D. Gur, J. H. Feist, F. L. Thaete, C. R. Fuhrman, C. A. Britton and B. S. Slasky, “Subjective and objective assessment of image quality- a comparison,” *Jour. Digit. Imaging* **7**, pp. 77-78, 1994.
13. K. M. Siddiqui, J. P. Johnson, B. I. Reiner, E. L. Siegel, “Discrete cosine transform JPEG compression vs. 2D JPEG2000 compression: JNDmetrix visual discrimination model image quality analysis,” *Proc. SPIE* **5748**, pp. 202-207, 2005.
14. T. Li, X. Li, J. Wang, J. Wen, H. Lu, J. Hsieh, and Z. Liang, "Nonlinear sinogram smoothing for low-dose x-ray CT," *IEEE Trans. Nuc. Sci.* **51**, pp. 2505-2513, 2004.
15. T. Lei and W. Sewchand, "Statistical approach to x-ray CT imaging and its application in image analysis," *IEEE Trans. Med. Imaging* **11**, pp. 53-61, 1992.
16. J. Nuyts, B. De Man, P. Dupont, M. Defrise, P. Suetens, and L. Mortelmans, "Iterative reconstruction for helical CT: a simulation study," *Phys. Med. Biol.* **43**, pp. 729-737, 1998.
17. A. Papoulis, *Random Variables and Stochastic Processes*. New York: McGraw Hill Book Company, NY, 1965.
18. P. A. Rattey and A. G. Lindgren, "Sampling the 2-D radon transform," *IEEE Trans. Acoust. Speech Signal Process.* **29**, pp. 994-1002, 1981.
19. J. Wang, T. Li, H. Lu, and Z. Liang, "Penalized weighted least-squares approach to sinogram noise reduction and image reconstruction for low-dose x-ray computed tomography," *IEEE Trans. Med. Imaging* **25**, pp. 1272-1283, 2006.
20. B. R. Whiting, "Signal statistics of x-ray CT," *Proc. SPIE Med. Imaging* **4682**, pp. 53-60, 2002.
21. A. Ziegler, T. Nielsen, and M. Grass, "Iterative reconstruction of a region of interest for transmission tomography," *Proc. SPIE Med. Imaging* **6142**, pp. 61421-23, 2006.
22. G. T. Herman, *Image Reconstruction from Projections*: Springer-Verlag Berlin Heidelberg, New York, NY, 1979.

Appendix G:

Development of continuous CT-guided minimally invasive surgery

Raj Shekhar[1,2,*], Omkar Dandekar[1,2], Steven Kavic[3], Ivan George[3], Reuben Mezrich[1], Adrian Park[3]

[1]Department of Diagnostic Radiology, University of Maryland, Baltimore, MD 21201, USA
[2]Department of Electrical & Computer Engineering, University of Maryland, College Park, MD 20742, USA
[3]Department of Surgery, University of Maryland, Baltimore, MD 21201, USA

ABSTRACT

Minimally invasive laparoscopic surgeries are known to lead to improved outcomes, less scarring, and significantly faster patient recovery as compared with conventional open invasive surgeries. Laparoscopes, used to visualize internal anatomy and guide laparoscopic surgeries, however, remain limited in visualization capability. Not only do they provide a relatively flat representation of the three-dimensional (3D) anatomy, they show only the exposed surfaces. A surgeon is thus unable to see inside a structure, which limits the precision of current-generation minimally invasive surgeries and is often a source of complications. To see inside a structure before dissecting it has been a long-standing need in minimally invasive laparoscopic surgeries, a need that laparoscopy is fundamentally limited in meeting. In this work we propose to use continuous computed tomography (CT) of the surgical field as a supplementary imaging tool to guide laparoscopic surgeries. The recent emergence of 64-slice CT and its continuing evolution make it an ideal candidate for four-dimensional (3D space + time) intraoperative imaging. We also propose a novel, elastic image registration-based technique to keep the net radiation dose within acceptable levels. We have successfully created 3D renderings from multislice CT corresponding to anatomy visible within the field of view of the laparoscope in a swine. These renderings show the underlying vasculature along with their latest intraoperative orientation. With additional developments, our research has the potential to help improve the precision of laparoscopic surgeries further, reduce complications, and expand the scope of minimally invasive surgeries.

Keywords: Image-guided surgery, laparoscopic surgery, augmented reality, elastic registration, 3D visualization

1. INTRODUCTION

Minimally invasive surgeries are a superior alternative to conventional open surgeries. In minimally invasive surgeries, the internal anatomy is accessed through a few small ports (holes) on the patient's skin rather than large incisions. The surgeon introduces the laparoscope (rigid endoscope) through one of the ports to illuminate the internal anatomy and uses the other ports to introduce surgical instruments. The region is often insufflated (filled) with CO_2 gas to make space for surgical manipulations and to provide access to the anatomy of interest. Minimally invasive surgeries performed under laparoscopic guidance have been shown to lead to improved outcomes, less scarring, and significantly faster patient recovery as compared to conventional open surgeries.[1] For certain surgical procedures, such as cholecystectomy (removal of gall bladder), minimally invasive surgery has become the standard of care.[2]

Despite the early success of minimally invasive surgeries, laparoscopes remain limited in visualization capability by their flat representation of three-dimensional (3D) anatomy and their ability to display only the most superficial surfaces. A surgeon is thus unable to see inside or around a structure, thus decreasing the precision of current-generation laparoscopic surgeries. Laparoscopic surgeons need awareness of the 3D operative field, especially visualization of the underlying blood vessels and other hidden structures.[3] Laparoscopes are fundamentally limited in providing this information and unable to meet this long-standing need.

[*] rshekhar@umm.edu; phone (410)706-8714; facsimile (410)706-8724

Medical Imaging 2007: Visualization and Image-Guided Procedures, edited by Kevin R. Cleary, Michael I. Miga,
Proc. of SPIE Vol. 6509, 65090D, (2007) · 1605-7422/07/$18 · doi: 10.1117/12.708783

Augmented reality (AR) has been suggested as a solution to overcome this limitation of laparoscopy. One approach to AR has been the creation of 3D models of organs from preoperative magnetic resonance (MR) or computed tomography (CT) images acquired days to weeks before the surgery.[4, 5] Such 3D visualization of anatomical structures from CT or MR imaging data is common in diagnostic radiology. Moreover, it is possible to expose hidden structures or to see inside organs by "peeling off" outer layers by making corresponding voxels transparent. These models have been subsequently rendered with exquisite detail and superimposed on real-time laparoscopics display for AR. Although this approach has shown the strength of combining CT or MR image-based visualization with laparoscopy, its accuracy remains suspect. The 3D organ models derived from preoperative CT or MR imaging are not current and do not represent the intraoperative anatomy that almost invariably will deform between the time of preoperative imaging and the surgical procedure.

We propose improving intraoperative visualization during laparoscopic surgeries through AR that uses 3D renderings of the anatomy scanned with live, intraoperative CT. Superimposition of such 3D views based on instantaneously acquired CT on the laparoscopic view after accounting for proper alignment has the potential to reveal hidden structures accurately and thereby assist the laparoscopic surgeon. Although computationally and practically more challenging, this approach does not suffer from the limitations of previously reported AR efforts. With the advent of 64-slice CT scanners, continuous intraoperative volumetric CT at high frame rates is becoming possible. The continual trend toward more slices (hence, volumetric coverage per rotation) and higher frame rate will make CT even more suitable for this surgical imaging task.

Radiation exposure to the patient and the surgeon is a concern with the use of continuous CT as proposed here. In this article, we not only present our preliminary results showing AR visualization using intraoperative CT but also describe a strategy to reduce the radiation dose based on registration of pre- and intraoperative CT. We conclude with a discussion of our results, strengths of our proposed strategy, and future directions of our research.

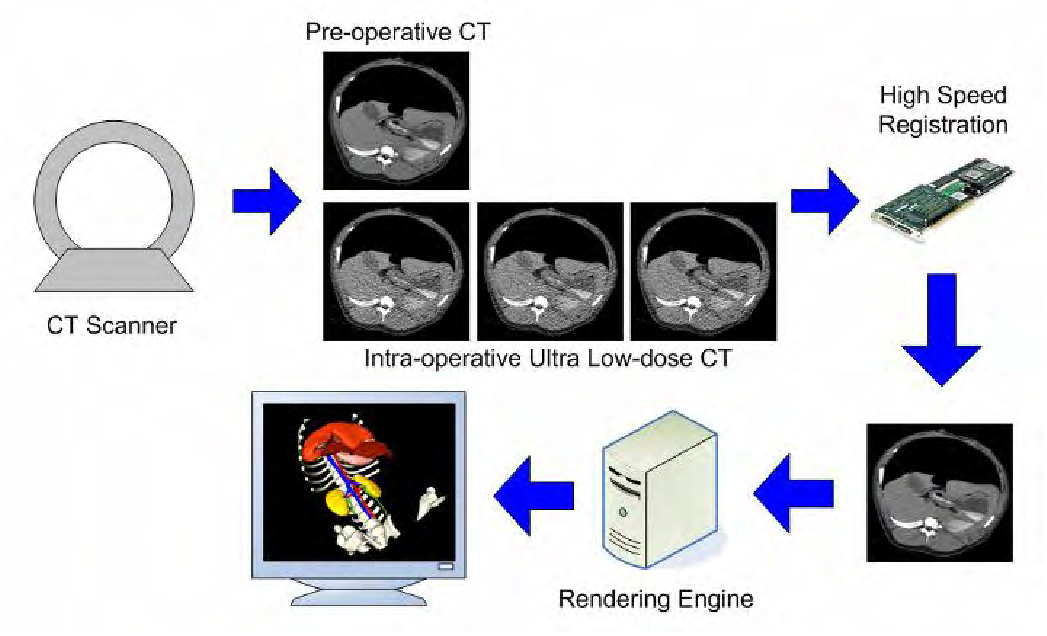


Figure 1 Flow diagram showing the CT protocol and various other steps involved in the proposed CT-guided laparoscopic surgery. Note that elastic image registration between standard-dose preoperative CT and ultra low-dose intraoperative CT helps reduce radiation exposure. A high-speed image registration engine in the final implementation will allow continuous intraoperative 3D visualization.

2. METHODS

1.1. Imaging protocol and dose reduction strategy

Figure 1 schematically describes our imaging protocol for the proposed CT-guided laparoscopic surgery and the concept behind our novel dose reduction strategy. After the surgery subject has been prepared (preparation includes insufflation) and immediately before surgery begins, we perform a contrast-enhanced CT scan at the standard diagnostic dose. The use of the contrast agent ensures that the desired blood vessels are highlighted in the CT. We call this initial 3D scan the preoperative CT scan. As soon as the surgery begins, we scan the dynamic operative field repeatedly with CT again, although this time the CT scanner is operated at a much lower dose. We refer to these subsequent nondiagnostic scans as intraoperative CT scans. Intraoperative CT scans are not contrast enhanced because of the short-acting nature of the CT contrast agents and the fact that these agents cannot be administered repeatedly without stress and potential harm to the kidneys and other critical organs.

Our next step is to elastically register pre- and intraoperative CT scans, which allows us to warp the diagnostic-quality preoperative scan in such a way that it matches the intraoperative anatomy. The warped preoperative CT scan, which has superior image quality, is then substituted for the intraoperative CT scan. This scan is subsequently rendered and superimposed on the corresponding laparoscopic view for AR. By repeating this process for each intraoperative CT scan several times per second, our approach can provide an accurate and up-to-date AR visualization throughout the surgery.

The dose reduction results from the use of ultra low-dose CT intraoperatively. Diagnostic CT scans are typically acquired at an x-ray tube voltage of 120 kVp and a tube current of 200 mAs. Working with a commercial CT dose simulator and archived patient images, we showed previously that accurate elastic image registration can be achieved even when the tube current is lowered to 10 mAs, thus resulting in an approximately 20-fold reduction in the radiation dose.[6] The dose reduction was slightly less when working with a commercial CT scanner because of preset limits on minimum tube current setting.

1.2. CT-laparoscopy spatial correlation

We performed the proposed continuous CT-guided laparoscopic surgery in a CT room working with a 64-slice scanner (Brilliance 64, Philips Medical System, Highland Heights, OH). Figure 2 shows our experimental setup. One of the first requirements for this novel surgical approach and AR visualization is to establish a spatial correlation between the CT coordinate system and the coordinate system of the laparoscope. Once initialized, the coordinate system of the CT scanner remains fixed. Because the laparoscope is manually operated, its coordinate system moves with it. We achieved the necessary spatial correlation through the use of an optical tracker (Polaris Spectra, Northern Digital, Waterloo, Canada). Infrared markers were attached to the external end of the laparoscope. A PC (termed the control PC) controlled the optical tracker, which, by tracking the infrared markers, provided the coordinates of the laparoscope in its (optical tracker's) coordinate space. A purpose-built calibration device with infrared markers visible in CT as well as visible to the optical tracker's cameras helped determine the transformation between the CT and the optical tracker coordinate systems.

The control PC was also fitted with a video frame grabber to capture and digitize the laparoscopic video. The control PC also synchronized the tracking and the video data. Temporal synchronization of laparoscopic data with the CT images is manual in our current system. Actions such as sudden movement of the laparoscope provided visual cues to synchronize CT and laparoscopy data.

1.3. Experimental protocol

In this preliminary study, our goal was to (1) prove the feasibility of accurate elastic registration between diagnostic preoperative CT and ultra low-dose intraoperative CT scans and the resulting radiation dose reduction, and (2) develop the necessary engineering tools to demonstrate AR visualization using warped preoperative CT. The focus was to achieve this goal for discrete time instants. By looping through these steps at a fast rate, the process can be made continuous in the future. Consequently, we followed the imaging protocol described in Figure 1, with the exception that continuous intraoperative CT was replaced with discrete single CT scanning. In addition, intraoperative CT and laparoscopy were performed sequentially while keeping the anatomic deformations to the minimum between these.

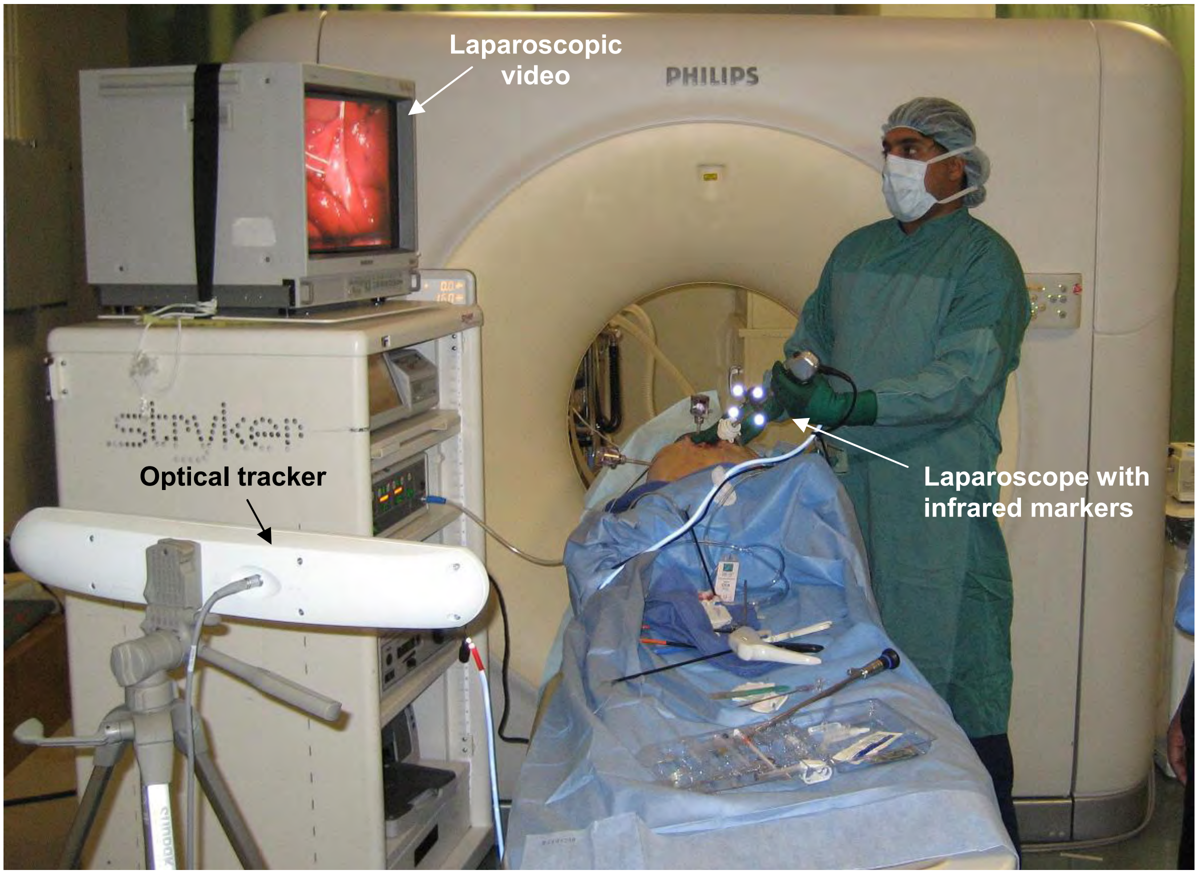


Figure 2 The setup for CT-guided minimally invasive laparoscopic surgery. The laparoscope is tracked using an optical tracker. The AR views, in this preliminary study, were created in an offline fashion.

We tested the proposed continuous CT-guided surgery approach on an anesthetized swine. Appropriate institutional approval was obtained for this animal protocol. Because we aim to demonstrate the feasibility of CT guidance for cholecystectomy (one of the most common minimally invasive surgeries) first, the particular organ of interest naturally was the gall bladder. The field of view for both CT and laparoscopy also included nearby organs such as the liver and the spleen. A preoperative CT scan with contrast-enhanced hepatic vasculature was acquired first at the standard dose per the protocol. A tube current setting of 100 mAs was used to account for preoperative CT to make adjustments for the smaller body weight of the animal. To simulate a surgical manipulation, the gall bladder was mobilized (i.e., separated from the liver and then elevated above the liver surface). Intraoperative CT scans were subsequently acquired at 4 different dose settings (100, 45, 30, and 15 mAs). Fifteen mAs was the lowest dose setting allowed by the scanner for the desired 1-mm CT. The x-ray tube voltage was kept fixed at 120 kVp.

1.4. Elastic image registration and AR visualization

The standard-dose preoperative CT scan and the low-dose intraoperative CT scans were registered as reported in our earlier dose simulator study using our own volume subdivision-based elastic image registration algorithm.[6,7] As before, the low-dose CT scans were preprocessed using an anisotropic diffusion filter. The registration was repeated for each of the four dose settings. The accuracy of registration was judged visually by evaluating the difference of warped preoperative CT and preprocessed intraoperative CT.

Independently, the organs and structures of interest (liver, gall bladder, and hepatic vasculature) in the preoperative CT scan were segmented using Amira 3D visualization software (Mercury Computers, Chelmsford, MA). The segmented structures were transformed according to the deformation field provided by elastic image registration with the intraoperative scan. The appropriately deformed segmented structures were surface rendered (again using Amira) from a series of viewing angles provided by the tracking data for the laparoscope. These rendered scenes were then superimposed with the corresponding laparoscopic video frames for AR visualization.

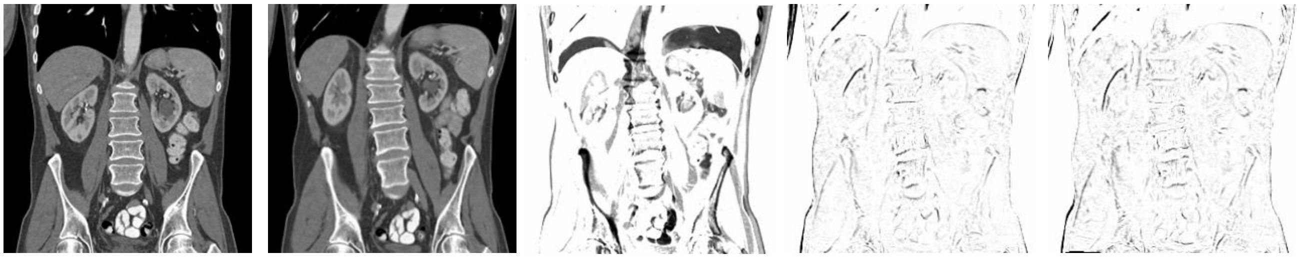
3. RESULTS

Our earlier simulation experiments, designed to measure the accuracy of nonrigid registration at various simulated doses, showed that intraoperative anatomical shifts can be tracked with an accuracy of 2 mm even at an x-ray tube current of 10 mAs (typical diagnostic imaging uses 200 mAs).[6] This is equivalent to a 94% and 95% reduction in the surface and deep tissue doses, respectively, indicating that continuous CT can provide safe and accurate surgical guidance. Figure 3 shows no visually noticeable difference in the performance of elastic image registration at high and low doses.

The same experiments were repeated for the swine data. In this case, the standard preoperative CT dose was lower (100 versus 200 mAs) to match the lower body weight of the animal compared with that of an adult human. In Figure 4, the pre- and intraoperative images are fused using a two-color scheme (red + blue channels for one image, green for another). The left panel shows the fused image before performing image registration. Bony structures such as the spine and ribs are clearly misaligned. The alignment improves after elastic image registration (middle and right panels). In this case, too, no visually noticeable difference in the registration quality is seen in images at the standard dose (100 mAs) or the lowest dose (15 mAs), indicating the potential of ultra low-dose intraoperative CT.

The procedures described above demonstrated the feasibility of substituting warped preoperative CT for intraoperative CT and reducing radiation exposure in the process. The warped preoperative CT is further used for high-quality 3D visualization and for creating a more accurate AR. Figure 5 displays side by side the laparoscopic and CT-generated views of the liver and the surrounding anatomy for a given time instant. Note that the hepatic vessels hidden beneath the liver surface are visible in the CT view. Laparoscopy cannot provide such information.

It is also important to note the benefit of registration for the visualization of the vasculature. Because the contrast agent cannot be used repeatedly, the vessels are not enhanced in intraoperative CT images. Even if the noise in the low-dose CT could be suppressed through a filtering operation, rendering intraoperative CT directly would not reveal the vasculature (see Figure 6, left panel). Only a few major vessels are visible here because of the residual CT contrast agent in them. These deep-seated vessels are not significant for a surgery like cholecystectomy. The contrast agent washes out fairly rapidly from the most peripheral vessels, which are of primary interest in most image-guided surgeries. Image registration allows use of the preoperative data for 3D visualization of the intraoperative anatomy while also retaining the vasculature information (see Figure 6, right panel).



Standard CT (Pose 1)

Standard CT (Pose 2)

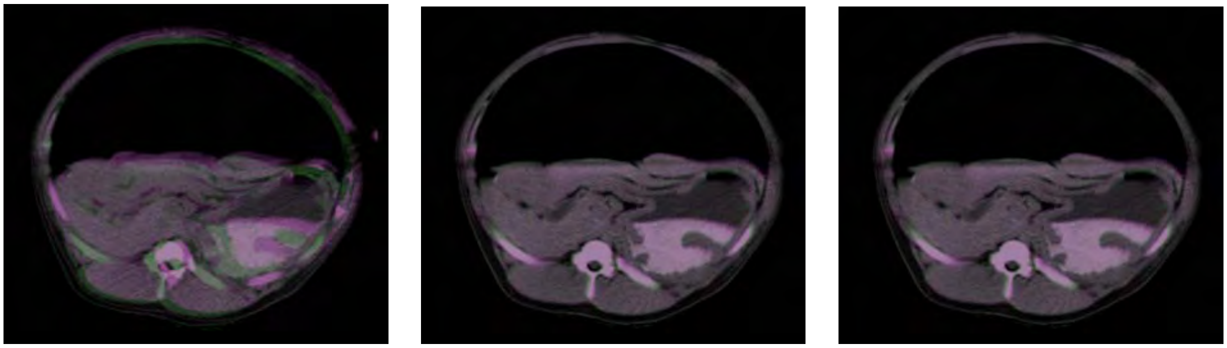
Starting Difference

200 mAs (Standard dose)

10 mAs (Ultra low dose)

Difference after elastic registration at standard dose and ultra low dose

Figure 3 Demonstration of elastic image registration accuracy for standard dose–low dose CT registration. The difference images suggest no visually noticeable difference in registration accuracy with dose. Quantitative data support this finding.



Before registration

After registration (100 mAs)

After registration (15 mAs)

Figure 4 Demonstration of elastic image registration accuracy for standard dose (100 mAs) – low dose (15 mAs) CT images of the swine. The difference images suggest no visually noticeable difference in registration accuracy even at a very low dose.

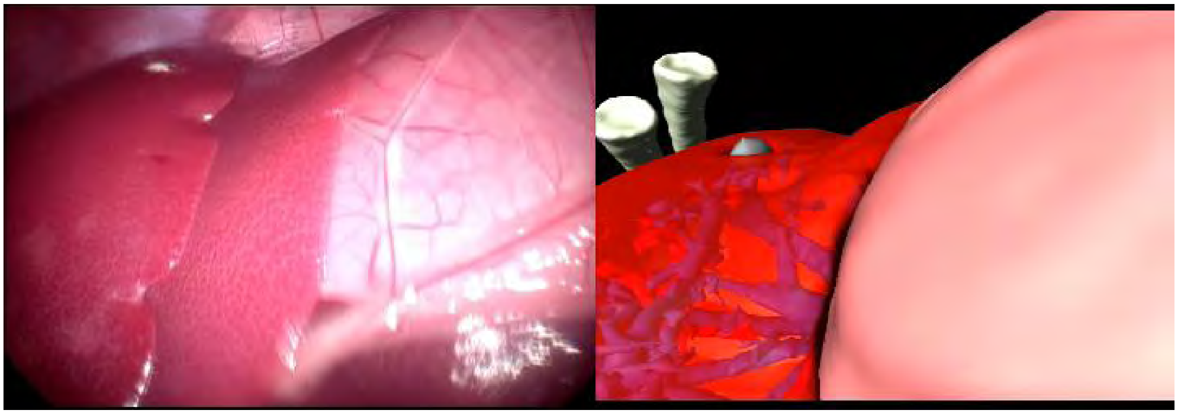


Figure 5 A comparison of spatially matching laparoscopic and CT-generated views. The CT-generated view (right panel) is capable of revealing the underlying vasculature, visualization of which is beneficial to laparoscopic surgeons.

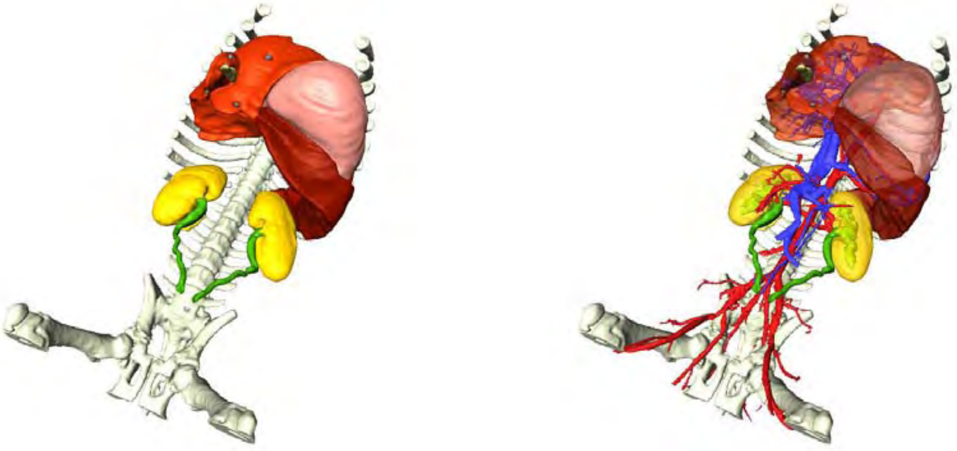


Figure 6 Because of the inability to reuse contrast material, a direct rendering of the preoperative CT (left), cannot show the detailed vasculature. The rendered warped CT (right) shows the vasculature.

Finally, we created AR views by blending semitransparent laparoscopic and CT-generated views. An example of such AR visualization using the instantaneous volumetric CT scan of the intraoperative anatomy is shown in Figure 7. Such an AR view preserves the surface texture information and optical depth cues from the laparoscopy while also exposing the underlying vasculature accurately.

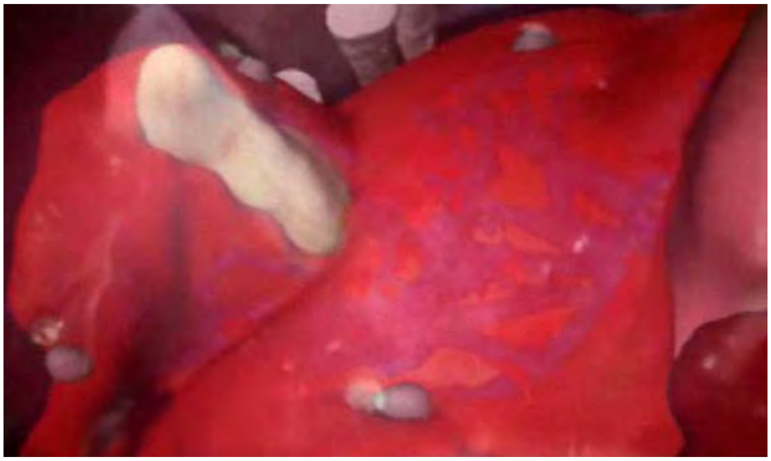


Figure 7 AR using 3D imagery from intraoperative CT. This novel AR technique preserves the surface texture information and optical depth cues from the laparoscopy while also exposing the underlying vasculature accurately.

4. DISCUSSION AND CONCLUSIONS

The described research is the first step toward an ambitious, long-term goal of taking advantage of volumetric imaging, now routine in diagnostic imaging, for minimally invasive surgeries. Laparoscopes, currently the primary visualization tool for navigating such surgeries, are limited in their 3D visualization capability. Essentially a video imaging technique, they cannot show structures under the exposed surfaces. AR, as proposed earlier, has provided the missing 3D information but is not accurate for extracranial surgeries, because the CT or MR imaging data employed for 3D visualization have not been current. We demonstrated the feasibility of using intraoperative data for AR here.

Our work combines real-time 3D imaging with minimally invasive laparoscopic surgery. Indeed, the greatest advantage of this approach is a dynamic 3D anatomical roadmap (rendering) to guide these procedures. Unlike the 3D roadmaps created in computer-assisted neurosurgery, our proposed 3D roadmap will refresh in real time to display tissue motion and surgical manipulations along with any surgical instruments within the operative field. Our approach is expected to initiate a new generation of minimally invasive surgeries relying on real-time 3D guidance. Incorporation of real-time 3D visualization and guidance is expected to allow laparoscopic surgeons to perform existing surgeries more precisely with fewer complications. Aided by improved visualization, it is also expected that many surgeries that are currently performed in an open invasive fashion can instead be performed minimally invasively, thereby reducing mortality and morbidity rates.

We have selected multislice CT as our intraoperative imaging modality for the proposed research. Any intraoperative imaging modality for guiding surgery must be volumetric and interactive (i.e., offer high frame rate). Current state-of-the-art 64-slice CT scanners provide a coverage of 4 cm, and we have been able to obtain 1 volumetric frame per second. Continuing advances in CT technology suggest that the coverage will grow to 10−12 cm. The potential exists for up to 8 volumetric frames per second speed as well. Therefore, evolving multislice CT technologies will become even more suitable for the proposed surgical application. In comparison, MR imaging lacks the speed needed for guiding interactive surgeries, and most surgical tools are not MR compatible. We have also found real-time 3D ultrasound unsuitable for the current application, because it cannot image across the pneumoperitoneum (caused by CO_2 insufflation). Cost and availability considerations also favor CT.

Radiation exposure to the patient and surgical team is a concern with the proposed continuous operation of the CT scanner. We showed up to 20-fold dose savings in an adult human through our strategy of elastically registering pre- and intraoperative CT images. Further savings may result if the CT scanners can be made to operate at even lower tube current setting. Visualization of critical underlying structures, especially the vasculature, is important before making surgical dissections. Inherent in our dose reduction strategy is a scheme to visualize the vessels without having to use the contrast continuously, which is neither permitted nor safe. Another advantage of having 3D models of anatomical structures in the CT-generated view is that one can interact with this view. For example, the surgeon can virtually practice a particular surgical manipulation and observe the effects of it in the CT view before actually making that manipulation. No such interaction is possible with the traditional laparoscopic view.

Engineering advances will be needed before continuous CT-guided laparoscopic surgery becomes routine. First, elastic image registration must be automatic and real-time. Our group has already made significant strides in this area, and we continue to improve the speed of elastic 3D image registration.[8.9] Full development of the proposed concept will also require a tight system-level integration among many subsystems and components: surgical tools, the laparoscope, tool tracking techniques, image processing (registration, in particular) technologies, visualization workstation, etc. Subsequently, a second level of integration will be required at the human/machine interface level that will combine the surgical protocol with the imaging protocol.

Continuous low-dose CT scanning of the dynamic operative field without exceeding acceptable radiation exposure and using high-speed elastic registration to generate diagnostic quality CT images of the intraoperative anatomy will enable high-quality 3D visualization of the operative field in a CT-equipped operating room. We have successfully created 3D renderings from multislice CT corresponding to anatomy visible within the field of view of the laparoscope. With additional developments, our research has the potential to help improve the precision of laparoscopic surgeries, reduce complications, and expand the scope of minimally invasive surgeries beyond its current 15% share of all surgeries.[2]

ACKNOWLEDGMENTS

This work was supported by the U.S. Department of Defense grant DAMD17-03-2-001.

REFERENCES

1. Himal HS. Minimally invasive (laparoscopic) surgery. *Surgical Endoscopy* 2002;16:1647-1652.
2. http://www.nibib1.nih.gov/events/IGI/IGIWorkshop2002_FINALReport.doc. 2002.
3. Harrell AG, Heniford BT. Minimally invasive abdominal surgery: lux et veritas past, present, and future. *Am J Surg* 2005;190:239-243.
4. Marescaux J, Rubino F, Arenas M*, et al.* Augmented-reality-assisted laparoscopic adrenalectomy. *Jama-Journal Of The American Medical Association* 2004;292:2214-2215.
5. Mutter D, Bouras G, Marescaux J. Digital technologies and quality improvement in cancer surgery. *Ejso* 2005;31:689-694.
6. Dandekar O, Walimbe V, Siddiqui K, Shekhar R, “Image registration accuracy with low-dose CT: How low can we go?”, In Proceedings of 2006 IEEE ISBI; p 502-505.
7. Walimbe V, Shekhar R. Automatic elastic image registration by interpolation of 3D rotations and translations from discrete rigid-body transformations. *Medical Image Analysis* 2006; 10:899-914.
8. Castro-Pareja CR, Jagadeesh JM, Shekhar R., “FAIR: A hardware architecture for real-time 3-D image registration”, *IEEE Trans Inf Technol Biomed* 2003;7(4):426-434.
9. Castro-Pareja CR, Shekhar R., “Hardware acceleration of mutual information-based 3D image registration”, *J Imaging Sci Technol* 2005; 49(2):105-113.

Appendix H:

Development of Continuous CT-Guided Minimally Invasive Surgery

Raj Shekhar, Omkar Dandekar, Steven Kavic, Ivan George, Reuben Mezrich, Adrian Park
University of Maryland, Baltimore; rshekhar@umm.edu

Minimally invasive surgeries performed under laparoscopic guidance lead to improved patient outcomes, less scarring and significantly faster patient recovery as compared to conventional open surgeries. Rigid endoscopes (laparoscopes) are used to visualize internal anatomy and guide laparoscopic surgeries. Laparoscopes, however, are limited in their visualization capability by their flat representation of three-dimensional (3D) anatomy and their ability to display only the most superficial surfaces. A surgeon is unable to see beneath visible surfaces, decreasing the precision of current-generation laparoscopic surgeries. Awareness of the 3D operative field is a long-standing need of laparoscopic surgeons that laparoscopes are fundamentally limited in meeting.

Our solution to this problem is to use continuous computed tomography (CT) of the operative field as a supplementary imaging tool to guide laparoscopic surgeries. 3D visualization of anatomical structures from CT data is common in diagnostic radiology. Moreover, it is possible to expose hidden structures or to see inside organs by “peeling off” outer layers by making corresponding voxels transparent. The recent emergence of 64-slice CT as well as its continuing evolution in speed and volumetric coverage makes it an ideal candidate for four-dimensional (3D space + time) intraoperative imaging. Cost and availability considerations and the ability to image across pneumoperitoneum (caused by CO_2 insuffulation) also favor CT.

Our initial attempts have focused on dose reduction and a preliminary demonstration of 3D visualization of the operative field using continuous CT. To minimize net radiation dose to the patient and the surgeon with the use of continuous CT, we have proposed a novel dose reduction strategy, in which we acquire a standard CT image preoperatively (following pneumoperitoneum) and scan the dynamic operative field using very low-dose CT once surgery begins. Using high-speed nonrigid 3D image registration (warping) techniques we have developed [1-2], we rapidly register the preoperative CT image to low-dose intraoperative CT images. Registered preoperative CT images, which match the intraoperative anatomy, are then substituted for the low-dose images, 3D rendered and presented to the surgeon.

Our simulation experiments, designed to measure the accuracy of nonrigid registration at various simulated doses, show that intraoperative tissue shifts can be tracked with an accuracy of 2 mm even at an x-ray tube current of 10 mAs (typical diagnostic imaging uses 200 mAs) [3]. This is equivalent to a 94% and 95% reduction in the surface and the deep tissue dose, respectively, indicating that continuous CT can provide safe and accurate surgical guidance.

In an early demonstration of CT-based intraoperative visualization, a swine, lying supine on the CT couch, was prepared for laparoscopic cholecystectomy as per the standard procedure. The laparoscope was tracked in space using an optical tracker, which was calibrated with the reference frame of the CT acquisition. The animation in Fig. 1 displays side-by-side the laparoscopic and CT-generated views of the liver and the surrounding anatomy. Note that the hepatic vessels hidden beneath the liver surface are visible in the CT view. Visualization of critical structures, such as the vasculature, is important before making surgical dissections.

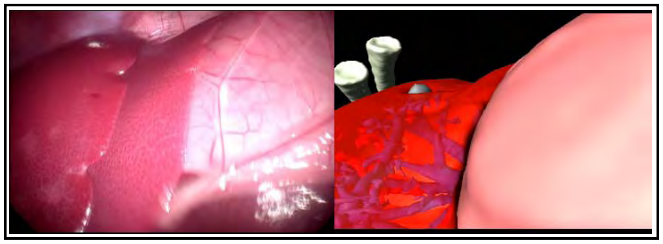


Figure 1 Laparoscopic display (left) versus the matched CT-generated view (right) [click for a movie]

Continuous low-dose CT scanning of the dynamic operative field without exceeding acceptable radiation exposure and using high-speed nonrigid registration to generate diagnostic quality CT images of the intraoperative anatomy will enable high-quality 3D visualization of operative field in a CT-equipped operating room (OR). We have successfully created 3D renderings from multi-slice CT corresponding to anatomy visible within the field of view of the laparoscope. We will extend this work to create augmented reality views as previously reported, albeit using instantaneous CT images. With additional developments, our research has the potential to help improve the precision of laparoscopic surgeries further, reduce complications, and expand the scope of minimally invasive surgeries to beyond its current 15% share of all surgeries.

References

1. Castro-Pareja CR, Jagadeesh JM, Shekhar R. FAIR: A hardware architecture for real-time 3-D image registration. *IEEE Trans Inf Technol Biomed* 2003;7(4):426-434.
2. Castro-Pareja CR, Shekhar R. Hardware acceleration of mutual information-based 3D image registration. *J Imaging Sci Technol* 2005; 49(2):105-113.
3. Dandekar O, Walimbe V, Siddiqui K, Shekhar R. Image registration accuracy with low-dose CT: How low can we go? In *Proceedings of 2006 IEEE ISBI*; p 502-505.